

CHARACTERIZATION OF ACOUSTIC EMISSIONS FROM STEEL
PIPELINE DENTS

A Thesis

by

JEFFREY LOUIS MICHEL

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee,	Peter Keating
Committee Members,	Stefan Hurlebaus
	Yong-Joe Kim
Head of Department,	Robin Autenrieth

August 2015

Major Subject: Civil Engineering

Copyright 2015 Jeffrey Louis Michel

ABSTRACT

Acoustic emissions monitoring is a novel form of nondestructive testing by which the elastic waves that propagate through materials and structures are detected and observed. It allows previously unknown information about the material to be acquired and analyzed. Deformation events often cause a sudden change in a material's stress field which is manifested as an elastic wave. Monitoring the emissions generated by deforming a material can lead to conclusions about the nature of its deformation that might otherwise go unseen. Acoustic activity varies based on the size, shape, and composition of the material, as well as the process by which the material is loaded. This study is an attempt to apply the methods of acoustic emissions monitoring to pipe dent tests, in which a dent is gradually induced into a steel pipeline segment. The purpose of these dent tests is to determine the critical strain at which the existence of a crack within the pipe is probable. Successful acoustic emissions monitoring would ideally be able to predict the generation of a crack and help sharpen the understanding of critical strain. An ultrasonic acoustic emissions monitoring system was acquired and applied to three pipe dent tests carried out over the course of this study. It was also used during a pipe burst test and steel tensile tests. All of these tests are analyzed using established characterization techniques. Some conclusions regarding the nature of the acoustic activity in the pipe dent tests are reached. Additionally, a number of general heuristics for the application of the acoustic emissions monitoring system are provided.

DEDICATION

This thesis is dedicated to my late grandmother, Phyllis Holowaty Albrecht, who passed away while it was written. You were unequaled in your enthusiasm for life, family, and community. You were my biggest fan and would have been thrilled when you heard I finished this paper. The world misses you.

ACKNOWLEDGEMENTS

I wish to express my gratitude to my research advisor, Dr. Peter Keating, for his guidance and support throughout this study. You are always calm and collected and I greatly respect the breadth of your knowledge and the ease with which you are able to confer it to your students. It was a pleasure to have worked with you.

I would like to acknowledge my other professors at Texas A&M, specifically Dr. Luciana Barroso and Dr. Stefan Hurlebaus, as well as Dr. Yong-Joe Kim for his assistance with this study. I would also like to thank my fellow graduate students who helped me along the way, specifically Kimberly Lyons, Patricia Escobar, Sam Hurley, Tim Barnes, Joshua McMahon, and Tristan Sarremejane. I wish you all the best!

Thank you to Matthew Potter and Ramiro Vanoye-Trevino who were instrumental in setting up and running the experiments in this study. Thanks for teaching me so much and dealing with my clumsiness!

I would also like to acknowledge Dr. Steven McCabe and the rest of the crew at the National Earthquake Hazards Reduction Program at NIST in Gaithersburg, MD. Working with you guys was such a great opportunity and I really appreciate everything you taught me. Steve, you've gone out of your way to help and support me this past year, and I consider myself lucky to be your friend and your colleague.

Special thanks Ana Gessel for all her love and support throughout my time here at Texas A&M. I'm so lucky to have met you and to have you in my life. I can't wait to start the next phase of our lives together.

I would like to thank all my friends and family who helped me along the way, specifically my brothers Stephen and Gregory Michel. Being so far away has made me appreciate even more how awesome you guys are, and I always greatly look forward to every opportunity we get to spend time together.

Most of all, I would like to acknowledge and thank my parents, Scott and Ann Michel. You are the smartest, most thoughtful, and generous people I know and I am so blessed by everything you have done for me over the course of my education and my life. I could not have done this without your love, support, and encouragement.

NOMENCLATURE

AE	Acoustic Emission
DAQ	Data Acquisition Device
HDT	Hit Definition Time
HI	Historic Index
HLT	Hit Lockout Time
IA	Intensity Analysis
LVDT	Linear Variable Differential Transformer
NDT	Nondestructive Testing
PDT	Peak Definition Time
PLB	Pencil Lead Break
SG	Strain Gauge

TABLE OF CONTENTS

	Page
ABSTRACT	ii
DEDICATION	iii
ACKNOWLEDGEMENTS	iv
NOMENCLATURE.....	vi
TABLE OF CONTENTS	viii
LIST OF FIGURES.....	ix
LIST OF TABLES	xiv
CHAPTER I INTRODUCTION	1
CHAPTER II BACKGROUND.....	3
2.1 Nondestructive Testing	3
2.2 Acoustic Emissions	3
2.2.1 Terminology	5
2.3 Ductile Fracture.....	11
2.3.1 Stages of Deformation.....	11
2.3.2 Acoustic Activity.....	12
2.4 Literature Review	14
2.4.1 Parameter Analysis.....	14
2.4.2 B-Value Analysis	16
2.4.3 Intensity Analysis	16
2.4.4 Cluster Analysis	18
CHAPTER III EXPERIMENTAL METHODS.....	21
3.1 Acoustic Emission Measurement	21
3.1.1 Pencil Lead Break Tests	25
3.1.2 Post-Processing	25
3.1.3 Transducer Mounts.....	30
3.2 Tensile Tests.....	34

3.3 Pipe Burst Test	37
3.4 Pipe Dent Tests.....	41
3.4.1 Dent Test 1	43
3.4.2 Dent Test 2	45
3.4.3 Dent Test 3	48
CHAPTER IV EXPERIMENTAL RESULTS	52
4.1 Transducer Tests	52
4.1.1 Sensor Calibration Tests.....	52
4.1.2 Software Parameters Tests	61
4.1.3 Wave Speed Test	65
4.1.4 Wave Attenuation Tests	65
4.2 Tensile Tests.....	69
4.3 Pipe Burst Test	72
4.4 Pipe Dent Tests.....	81
4.4.1 Dent Test 1	81
4.4.2 Dent Test 2	92
4.4.3 Dent Test 3	99
CHAPTER V SUMMARY AND CONCLUSIONS	121
5.1 Dent Test Conclusions	121
5.2 Acoustic Emissions Conclusions and Recommendations	123
5.3 Directions for Future Research.....	125
REFERENCES.....	127
APPENDIX A MATLAB© CODES	131
A.1 – AEAnalysis.M	131
A.2 – SETTest.m.....	146
A.3 – BurstTestExcelCorrector.m	153
A.4 – Pipe 9ExcelCorrector.m	154
A.5 – Pipe 4ExcelCorrector.m	157
A.6 – Pipe 2ExcelCorrector.m	159
A.7 – CreateAEdata.m	161
A.8 – CreateBin.m.....	162
A.9 – LoadWaveform.m.....	163
A.10 – DataExtraction.m.....	166
A.11 – IHitsVector.m	168
A.12 – IntensityAnalysis.m.....	169
A.13 – NormalizeVector.m	170

LIST OF FIGURES

	Page
Figure 2-1: Continuous and Burst Type AE Signals	6
Figure 2-2: Typical AE Characteristics	6
Figure 2-3: Kaiser and Felicity Effects	7
Figure 2-4: Typical AE Monitoring Setup	7
Figure 2-5: Stress-Strain Relationship of Steel in Tension	12
Figure 2-6: Typical Relationship between Steel Deformation and AE Activity	13
Figure 2-7: Intensity Chart Example	18
Figure 2-8: Example of Data Clusters	19
Figure 2-9: K-NN and K-Means Cluster Centers	20
Figure 3-1: AE Monitoring System	21
Figure 3-2: Transducers, Magnetic Mount, and Couplant	22
Figure 3-3: MISTRAS Data Acquisition Device	23
Figure 3-4: AEWIn [™] Software Screenshot	24
Figure 3-5: Justification of Post-Processing Floating Thresholds	27
Figure 3-6: Justification of Additional Post-Processing HDT	28
Figure 3-7: Justification of Post-Processing Power Spectrum Limits	29
Figure 3-8: Original Transducer Mount on Dent Pipe	31
Figure 3-9: Example of Improper Sensor-to-Surface Contact	32
Figure 3-10: New Transducer Mount	33
Figure 3-11: MTS Tensile Testing Machine	35

Figure 3-12: Tensile Test Sensors	36
Figure 3-13: Burst Test – Pipe with Pressurization Setup	37
Figure 3-14: Burst Test – Pipe with Protective Shielding	38
Figure 3-15: Burst Test – Transducer Next to Defect	39
Figure 3-16: Burst Test – Post Failure	40
Figure 3-17: Denting Frame with Dent Pipe Underneath	42
Figure 3-18: Dent Test 1 – Pre Dent, Pipe underneath Denting Frame	44
Figure 3-19: Dent Test 1 – View of Final Dent through Pipe	44
Figure 3-20: Dent Test 2 – Pre Dent, Pipe underneath Denting Frame	46
Figure 3-21: Dent Test 2 – Initial Crack inside Pipe at Dent Apex	46
Figure 3-22: Dent Test 2 – Final Crack inside Pipe at Dent Apex	47
Figure 3-23: Dent Test 3 – Pre Dent, Pipe underneath Denting Frame	48
Figure 3-24: Dent Test 3 – Initial Crack inside Pipe at Dent Apex	49
Figure 3-25: Dent Test 3 – Final Crack inside Pipe at Dent Apex	49
Figure 3-26: Dent Test 3 – Transducer Placement next to Indenter	50
Figure 3-27: Dent Test 3 – Transducer Positions at Final Dent Depth	51
Figure 4-1: Calibration Test 1 Waveforms	54
Figure 4-2: Calibration Test 1 Comparison	55
Figure 4-3: Calibration Test 2 Waveforms	57
Figure 4-4: Calibration Test 2 Comparison	58
Figure 4-5: Calibration Test 3 Waveforms	60
Figure 4-6: Calibration Test 3 Comparison	60
Figure 4-7: Analog Filter Frequency Spectra	63
Figure 4-8: Waveform with Low Frequency Content	64

Figure 4-9: Waveform Attenuation Parameters	66
Figure 4-10: Waveform Attenuation Examples, Short.....	67
Figure 4-11: Waveform Attenuation Examples, Long.....	68
Figure 4-12: Tensile Test – Low Acoustic Activity.....	69
Figure 4-13: Tensile Test – High Acoustic Activity	70
Figure 4-14: Tensile Test – API X52 Activity	71
Figure 4-15: Burst Test – Pressure vs. Amplitudes.....	73
Figure 4-16: Burst Test - Cumulative Hits vs. Pressure.....	74
Figure 4-17: Burst Test – Cumulative Hits vs. Pressure (Zoomed)	74
Figure 4-18: Burst Test – Historic Index and Severity	75
Figure 4-19: Burst Test – Amplitudes vs. Peak Frequencies	76
Figure 4-20: Burst Test – Amplitudes vs. Frequency Centroids.....	76
Figure 4-21: Burst Test – Amplitudes vs. Durations	77
Figure 4-22: Burst Test – Amplitudes vs. Rise Times	77
Figure 4-23: Burst Test – Durations vs. Rise Times	78
Figure 4-24: Burst Test – Types of Large Duration Hits	80
Figure 4-25: Dent Test 1 – Cumulative Hits	82
Figure 4-26: Dent Test 1 – Depths vs Amplitudes.....	83
Figure 4-27: Dent Test 1 – Instrumentation	84
Figure 4-28: Dent Test 1 – Strongest Signal	85
Figure 4-29: Dent Test 1 – Cumulative Signal Strength	86
Figure 4-30: Dent Test 1 – Historic Index and Severity	86
Figure 4-31: Dent Test 1 – Amplitudes vs. Peak Frequencies	87
Figure 4-32: Dent Test 1 – Amplitudes vs. Frequency Centroids.....	88

Figure 4-33: Dent Test 1 – Amplitudes vs. Durations	88
Figure 4-34: Dent Test 1 – Amplitudes vs. Rise Times	89
Figure 4-35: Dent Test 1 – Durations vs. Rise Times	89
Figure 4-36: Dent Test 1 – Depths vs. Amplitudes for Sharp Hits	91
Figure 4-37: Dent Test 2 – Cumulative Hits	92
Figure 4-38: Dent Test 2 – Depths vs Amplitudes	93
Figure 4-39: Dent Test 2 – Instrumentation	93
Figure 4-40: Dent Test 2 – Cumulative Signal Strength	94
Figure 4-41: Dent Test 2 – Historic Index and Severity	95
Figure 4-42: Dent Test 2 – Amplitudes vs. Peak Frequencies	95
Figure 4-43: Dent Test 2 – Amplitudes vs. Frequency Centroids	96
Figure 4-44: Dent Test 2 – Amplitudes vs. Durations	96
Figure 4-45: Dent Test 2 – Amplitudes vs. Rise Times	97
Figure 4-46: Dent Test 2 – Durations vs. Rise Times	97
Figure 4-47: Dent Test 3 – Cumulative Hits	100
Figure 4-48: Dent Test 3 – Depths vs. Amplitudes	101
Figure 4-49: Dent Test 3 – Instrumentation	102
Figure 4-50: Dent Test 3 – Cumulative Signal Strength	103
Figure 4-51: Dent Test 3 – Historic Index and Severity	103
Figure 4-52: Dent Test 3 – Historic Index vs. Severity	104
Figure 4-53: Dent Test 3 – Amplitudes vs. Peak Frequencies	105
Figure 4-54: Dent Test 3 – Amplitudes vs. Frequency Centroids	105
Figure 4-55: Dent Test 3 – Sensor 1 Low Frequency Peak, Weak Hit	106
Figure 4-56: Dent Test 3 – Sensor 1 Low Frequency Peak, Strong Hit Comparison	107

Figure 4-57: Dent Test 3 - Low Frequency Peak, Strong Hit and Reflection	108
Figure 4-58: Dent Test 3 – Sensor 2 Low Frequency Peak, Strong Hit Comparison	109
Figure 4-59: Dent Test 3 – Depths vs. Amplitudes, Low Frequency Peaks	110
Figure 4-60: Dent Test 3 – Amplitudes vs. Durations	110
Figure 4-61: Dent Test 3 – Amplitudes vs. Rise Times	111
Figure 4-62: Dent Test 3 – Durations vs. Rise Times	111
Figure 4-63: Dent Test 3 – Depths vs. Durations.....	112
Figure 4-64: Dent Test 3 – Sensor 2 Long Rise Time Hit	113
Figure 4-65: Dent Test 3 – Sensor 2 Dual Hit.....	114
Figure 4-66: Dent Test 3 – Sensor 2 Large Dual Hit	115
Figure 4-67: Dent Test 3 – Sensor 2 Large Dual Hit Comparison.....	116
Figure 4-68: Dent Test 3 – K-Means Depths vs. Amplitudes	118
Figure 4-69: Dent Test 3 – K-Means Amplitudes vs. Durations	119
Figure 4-70: Dent Test 3 – K-Means Durations vs. Rise Times	119

LIST OF TABLES

	Page
Table 3-1: Tensile Test Chemical Analysis	34
Table 3-2: Dent Test Physical Properties	41
Table 3-3: Dent Test Chemical Analysis	41
Table 4-1: Calibration Test 1 Parameters	53
Table 4-2: Calibration Test 2 Parameters	56
Table 4-3: Calibration Test 3 Parameters	59
Table 4-4: Software Parameter Tests	62

CHAPTER I

INTRODUCTION

Pipe dent tests are part of an ongoing research program involving critical at Texas A&M University's High Bay Structural and Materials Testing Laboratory. These tests involve gradually pressing mechanical dents into decommissioned steel pipeline segments, preferably until a crack has appeared in the dent area. A laser scanning system is used to measure pipe body strains accurately which are then used to increase the accuracy of finite element analyses of similar dent formations. The purpose of this research program is to determine a critical strain factor which can determine the potential existence of a crack in pipeline dents discovered in the field.

Sometimes during these pipe dent test, metallic "pings" are heard at advanced stages of dent deformation. This phenomenon suggested the possibility that acoustic emissions monitoring could provide useful information about dent deformation and crack growth. Acoustic emissions generated by plastic deformation events in the dent could theoretically affirm the existence of unseen microcracks within the steel. Ideally, acoustic sensors could even pick up the micromechanical events that precede crack generation. The ability to predetermine when a crack in a dent test is about to appear would be a great boon to the advancement of this critical strain research.

The hardware and software necessary for the observation of the ultrasonic acoustic emissions generated by deformation events within steel was acquired. Soon

after, it was tested during a pipe burst test conducted for a separate research regimen. It was then used for three pipe dent tests that coincided with this study. Additionally, the monitoring system was used during steel tensile tests in an attempt to correlate them to the dent tests. Analysis of all these tests are included in this study.

Since the Texas A&M University High Bay Laboratory did not previously have the capability to measure acoustic emissions, the research conducted in this study not only represents an attempt to further the development of the critical strain research, but an introduction to the possibilities and potential pitfalls of acoustic emissions monitoring during the testing of other structures and materials. Familiarity with the acoustic emissions monitoring system grew over the course this study. As a result, these tests represented learning opportunities to further understand the nature of the relationship between deformation event, physical waveform, analog transducer response, and digital signal. While certain conclusions are attainable from the analyses to follow, this study primarily serves as an introduction the field of structural acoustics monitoring and a compass for further research and applications.

CHAPTER II

BACKGROUND

2.1 NONDESTRUCTIVE TESTING

Nondestructive testing (NDT) has been defined as the set of methods for testing structures and materials which does not impair their usefulness or integrity [1]. These methods include visual inspection, liquid penetrant testing, magnetic particle inspection, radiographic testing, infrared and thermal testing, and acoustic emissions testing [1]. NDT is typically used to detect and evaluate flaws or damage in components while they are in use. It is also advantageous in laboratory experiments because it can provide additional monitoring of test specimen without major interference with the test itself.

2.2 ACOUSTIC EMISSIONS

Acoustic emissions (AE) are defined as the phenomena by which stress waves are generated by the rapid release of energy from localized sources within a material [2]. Familiar examples include the cracking of wood and the crumpling of foil. The scope of AE is not limited to audible events (which is typically taken to be 20Hz-20kHz), but include all phenomenon by which energy is propagated through a medium by a displacement wave, including earthquakes and shock waves. In general, whenever a mechanical phenomenon causes a sudden change in a material's stress field, that change

is manifested by a radiating stress wave, and that stress wave is interpreted as an acoustic emission. In metals, the localized sources of AE include but are not limited to crack initiation, crack propagation, phase transitions, deformation twinning, dislocation motion, fracture of inclusions, and interior friction [3]. The source event determines the initial characteristics of the AE waveform such as amplitude and frequency content [4, 5]. As the wave travels, these characteristics are altered by the geometry and material properties of the medium [6]. Exterior sources such as background noise and friction between objects also produce measurable AE and must be identified and accounted for in testing.

There are two general categories of AE test methodologies: live monitoring and induced acoustic testing. Live monitoring involves collecting acoustic data generated by a material or component during its use. This method is used to observe previously unseen activity within the test subject and collect useful data about its behavior. For example, live acoustic monitoring is often used to examine the propagation of fatigue cracks in steel. Induced acoustic testing involves generating and monitoring an acoustic wave as it passes through a medium. The behavior of the sound wave inside the test material can provide insights into its internal structure. This method is often used by inspectors to detect voids inside concrete structures and weld flaws during steel fabrication.

Material acoustic activity has been observed and used throughout history. Craftsmen, potters, and builders of all sorts learned to tap and listen to their creations to learn more about their structural integrity. Early engineers and metallurgists made

observations about audible noises generated during the deformation of tin. But the field of material acoustics did not enter the literature until the beginning of the 20th century, when a number of independent studies on AE started to arise. The first major study, which was the beginning of modern AE research, was carried out by Josef Kaiser in 1950 [3]. Kaiser was the first to observe and record the noises generated by a variety of engineering materials during deformation and correlate them to the stress-strain curve. He paved the way for a number of other AE studies in the second half of the 20th century, and is known for the Kaiser Effect for which he is named.

Today, AE studies is a fully realized but continually growing field in science and engineering. Advances in electronics and computer processing have allowed for accurate monitoring and storage of acoustic data. Researchers from a wide range of disciplines including geology, medicine, and engineering use acoustic techniques to help them see the unseen and advance their knowledge of the internal structure and behavior of their subjects.

2.2.1 Terminology

There are two types of acoustic emissions: continuous and burst signals, shown in the figures below. Continuous signals are usually created by background noise while burst signals represent a sudden release of elastic energy. Most measured burst signals contain an underlying continuous signal that can either be filtered out or ignored if the burst is strong enough.

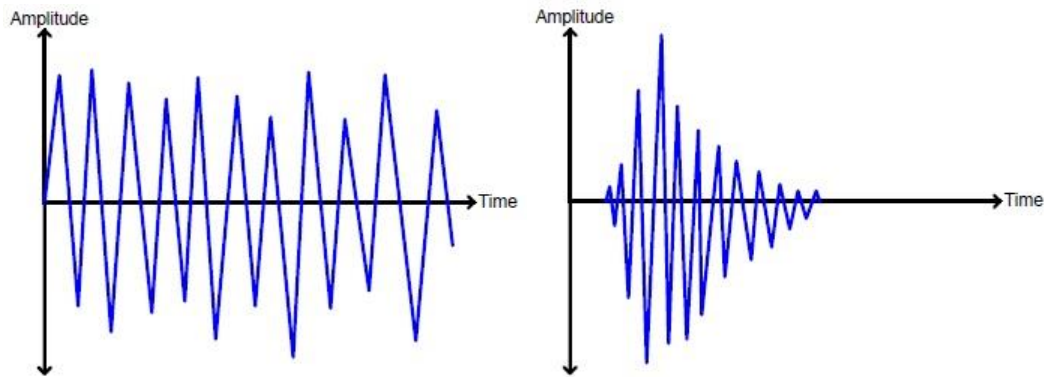


Figure 2-1: Continuous and Burst Type AE Signals

There is a fair amount nomenclature associated with AE which are defined in the following pages, with some accompanying figures. Figure 2-2 provides a graphical representation of some terms associated with a burst type AE waveform. Figure 2-3 illustrates some common AE phenomena. Figure 2-4 shows a typical setup for an AE monitoring test.

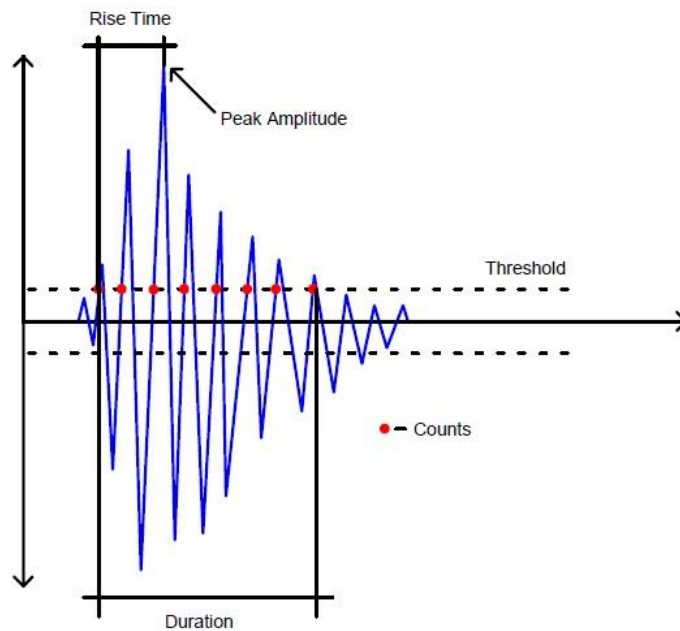


Figure 2-2: Typical AE Characteristics

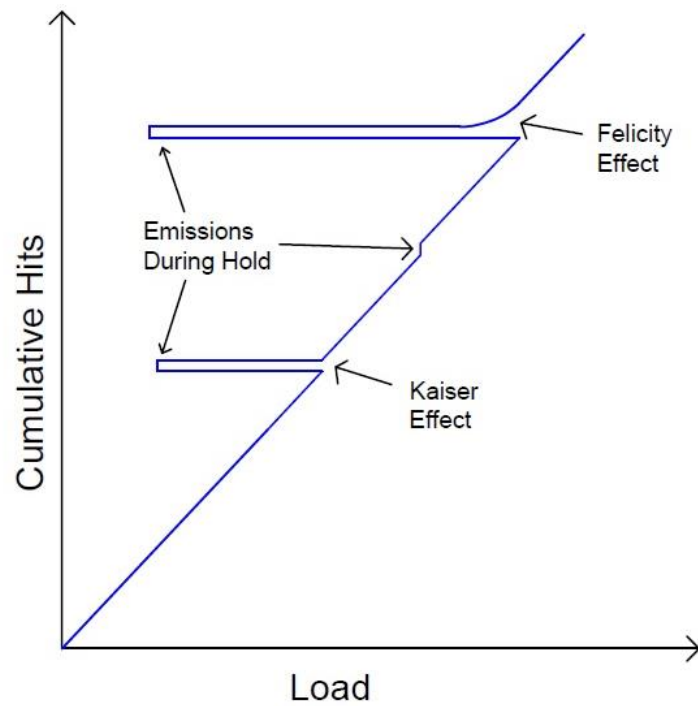


Figure 2-3: Kaiser and Felicity Effects

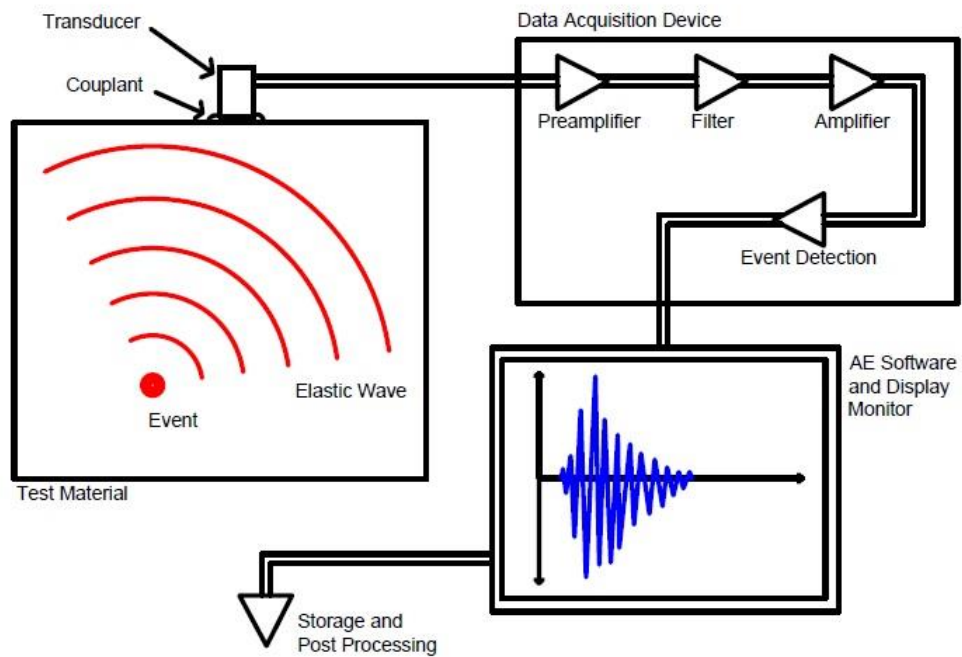


Figure 2-4: Typical AE Monitoring Setup

Event – A name given to an acoustic emission received by the sensor. It can be used to refer to the emission itself, or the source of the emission.

Hit – Another name given to an acoustic emission received by the sensor, typically a burst type event.

Amplitude – The magnitude of a signal received by the sensor. While amplitude may be used to reference the magnitude at any point on the waveform, typically the maximum amplitude of a waveform is just referred to as its amplitude. Amplitude is either expressed in volts or in decibels, which are related by the following equation.

$$dB = 20 * \log_{10} \left(\frac{V}{0.01} \right) + Gain \quad (1)$$

Gain - A linear boost in signal applied by the pre-amplifier during waveform acquisition.

Threshold – AE software is generally programmed to record any signal which crosses a certain threshold amplitude. Sometimes floating thresholds are used, which are functions of the signals themselves.

Count – Each event has a certain number of counts, defined as the amount of times the signal crosses the detection threshold. Generally each oscillation that crosses the positive and negative thresholds is considered a single count.

Rise Time – The amount of time between the first count and the peak amplitude.

Duration – The amount of time the first count and the last count.

Ring-down Time – The difference between the rise time and the duration.

Power Spectrum – The power spectrum is the plot of the frequencies contained in the waveform along with their corresponding excitation magnitudes. It is obtained by performing a Fourier transform on the event waveform, which is an algorithm that represents the waveform as a series of sine functions with discrete frequencies [7].

Peak Frequency – An event's maximum excited frequency as shown by the power spectrum.

Frequency Centroid – The “center of mass” of the power spectrum, which is used as a representation of the distribution of frequencies. It is defined as a weighted average of the frequencies contained in the power spectrum.

Initiation Frequency – The ratio of the counts until peak and rise time.

Reverberation Frequency – The ratio of the counts after peak and ring-down time.

Kaiser Effect – A acoustic phenomenon whereby a material that is loaded, then unloaded, and then loaded again will produce AE during the first loading, but not during the second loading until the stress exceeds the maximum experienced in the first loading [2, 8].

Felicity Effect – An exception to the Kaiser Effect whereby load reversals above a certain stress intensity will still produce AE during reloading [2].

Transducer – A sensor which converts mechanical energy into electrical energy.

Acoustic transducers contain a piezoelectric crystal which creates a voltage when stress is induced. The transducer is the key component in any AE monitoring system as it represents the boundary between the physical acoustic wave and the digital waveform.

Couplant – A viscous material used to improve the transition of the acoustic wave from the material to the transducer.

Data Acquisition Device (DAQ) – A processing board capable of converting the analog voltage input from the transducer into a digital signal recognizable by the computer software.

Preamplifier/Amplifier – Since the signal from the transducer is generally very weak, amplifiers are used to boost it before and after the analog filter is applied.

Filter – The DAQ contains an adjustable analog filter that defines a range of passable frequencies. Additional filters may be applied during post-processing to separate hits with a more selective band of frequencies.

Peak Definition Time (PDT) – A software parameter that ensures a correct definition of peak amplitude. Since waveform peaks are all relative, the software waits to label a local maximum as the peak amplitude until the PDT has passed and no greater peak is detected.

Hit Definition Time (HDT) – A software parameter that helps determine the end of a hit. Once a waveform dips back below the detection threshold, the software will wait for the HDT before declaring the end of the hit. Since waveforms have a tendency to display multiple bursts, this prevents the software from defining a single hit as multiple hits.

Hit Lockout Time (HLT) – A software parameter that inhibits the detection of waveform reflections by preventing detection for a set amount of time after the HDT expires.

2.3 DUCTILE FRACTURE

Ductile fracture, also referred to as rupture, is a failure mode of steel which is accompanied by large plastic deformations. Intense strains lead to the formation of microvoids within the steel which coalesce and eventually form a crack [9]. In a steel tensile test this causes a sudden fracture event. In the pipe dents explored in this study, the material reaches a critical strain at a certain dent depth after which there is a possibility that an interior crack has manifested. Sometimes the cracks occur on the pipe surface and their propagation can be directly observed.

2.3.1 Stages of Deformation

Typically the deformation of steel is classified into elastic and plastic regions. Elastic deformation is defined as the region when stress and strain are proportional, with the constant of proportionality defined as the modulus of elasticity [9]. This proportional relationship is known as Hooke's law. Elastic deformation is reversible, which means that the material will revert to its original shape after unloading. Macroscopic elastic strain is manifested as a gradual stretching of interatomic bonds [9].

Plastic deformation occurs at higher stresses than elastic deformation and represents irreversible deformation [9]. Interatomic bonds break and reform and do not revert to their original state when stress is relieved [9]. Stress and strain are no longer directly proportional. In a typical tensile test, the steel will continue to strain after yield

while the load holds steady until the onset of strain hardening. The load will then gradually increase to a maximum and eventually lead to fracture. Figure 2-5 shows a typical relationship between stress and strain for steel in tension. It shows the engineering stress and strain, which is calculated from a specimen's original cross section, as opposed to the true stress and strain which are calculated from the changing cross section.

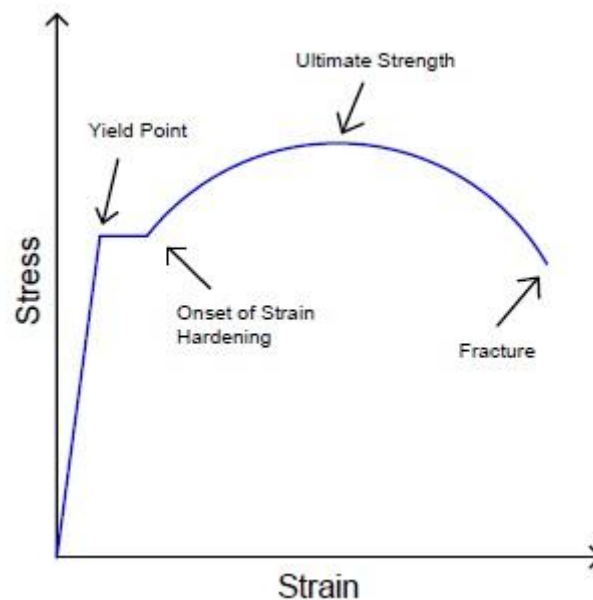


Figure 2-5: Stress-Strain Relationship of Steel in Tension

2.3.2 Acoustic Activity

Typically the deformation events in ductile fracture that generate burst type acoustic emissions are dislocation movement, stress-induced phase transformations, and

the phenomena associated with crack growth such as microvoid nucleation and coalescence [3, 10, 11]. A dislocation occurs when molecular bonds within the steel break and reform resulting in a permanent realignment of the crystalline structure [9]. A phase transformation is a change in the chemical characteristics of a portion of the microstructure of an alloy such as steel [9]. Eventually, microvoids are formed within the steel as a result of high local strains overcoming intermolecular bonds [9]. As more voids appear they tend to coalesce into microcracks which can expand into macrocracks.

The acoustic activity associated with the deformation of steel varies depending on the geometry of the test specimen, the nature of the induced stress, and the specific failure mechanisms. Typically, in a basic tensile test, acoustic activity reaches its peak after the steel yields and gradually drops off before fracture as shown in Figure 2-6 [12, 13].

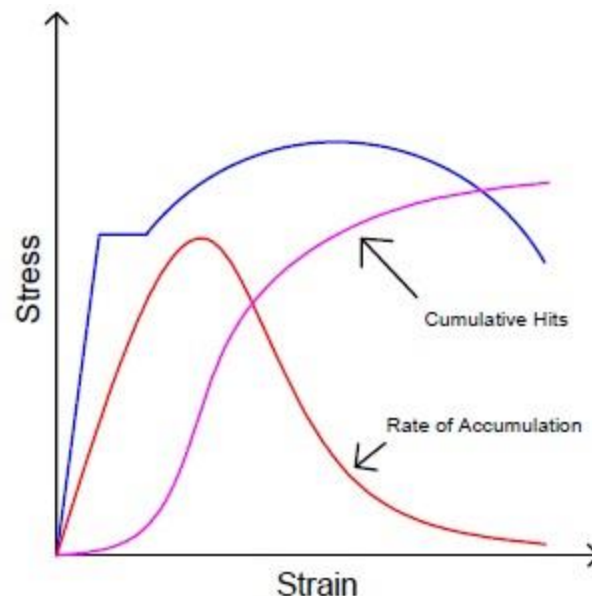


Figure 2-6: Typical Relationship between Steel Deformation and AE Activity

2.4 LITERATURE REVIEW

Acoustic emissions technology has been used to research and analyze damage mechanisms in a variety of structural materials. It has been more commonly used to monitor reinforced concrete or composite systems, but these analyses can still be helpful for AE monitoring of other materials [14, 15, 16, 17, 18, 19]. As for steel, the majority of AE monitoring is conducted for fatigue fracture tests [11, 20, 21, 22, 23, 24, 4, 25]. There have been some studies that use AE for steel pipelines [26, 27, 28, 29]. And there are a number of studies that involved monitoring AE activity during tensile tests [30, 10, 31, 13, 12]. All of these studies provide useful insight into the detection and characterization of AE signals measured during material deformation. What follows is a description of some of the more common characterization techniques that may prove useful for the tests contained in this study.

2.4.1 Parameter Analysis

Parameter analysis involves using specific properties of individual acoustic events—amplitude, frequency, duration, etc.—to classify and characterize them. All AE studies use parameter analysis in some form. Amplitude is generally the most common parameter used in analysis because a signal's amplitude is a good indication of the severity of the source event. Signal duration can also sometimes indicate the type of source event or the distance between the sensor and the source. Burst events are often

defined as signals with relatively short durations and large amplitudes, while signals with longer durations and shorter amplitudes are usually associated with continuous noise. This is an example of a parameter analysis can be used to filter unwanted signals during post-processing [25].

An AE signal's energy content, as defined by the equation below, is another way to measure its intensity [32].

$$E_i = \int_{t_0}^{t_1} V_i(t)^2 dt \quad (2)$$

E_i = Energy content.

$V_i(t)$ = Signal waveform.

t_0, t_1 = Beginning and end of signal.

Similarly, a signal's strength is defined as the integral of the rectified (absolute value) signal voltage [3]. The cumulative signal strength (CSS) is considered a useful metric for damage assessment [14, 18]. If CSS obtained during a test is plotted against the load or stress, spikes and troughs in that curve can sometimes indicate when significant internal damage has occurred. Since signal energy and signal strength are functions of the sensor signal and not the waveform itself, they are not direct indications of event source energy and are generally only used to relate signals to each other.

Frequency content can sometimes be useful in analysis, however a signal's frequency content is one of the parameters that is most affected by material and transducer properties [27].

2.4.2 B-Value Analysis

The b-value analysis is a probabilistic method that uses the hits measured in a previous test to predict the hits measured in future tests [14, 20]. It was originally developed by Gutenberg and Richter to estimate the likelihood of earthquakes above a certain magnitude [14], but has been translated to AE research to help predict acoustic activity. It uses the following logarithmic relation.

$$\log(N) = C - b \log(A) \quad (3)$$

N = Number of AE hits up to time t.

A = Threshold amplitude.

C = Numerical constant.

b = Derived constant to relate N and A.

Once a b-value has been established for a specific material under a given loading condition with a known outcome, this b-value analysis can be used for future similar tests to ascertain whether mechanical damage is greater or less than expected.

2.4.3 Intensity Analysis

Intensity analysis (IA) evaluates the significance of AE events as well as the damage sustained by the material by calculating the historic index (HI) and the severity (S_r) at various times, given by the following equations [33].

$$HI = \frac{N}{N - K} \left(\frac{\sum_{i=K+1}^N S_{oi}}{\sum_{i=1}^N S_{oi}} \right) \quad (4)$$

$$S_r = \frac{1}{J} \left(\sum_{m=1}^J S_{om} \right) \quad (5)$$

N = Number of AE hits up to time t.

S_{oi} = Signal strength of the i th hit, ordered by detection time.

S_{om} = Signal strength of the m th hit, ordered by magnitude.

K, J = Constants based on the material and the number of hits. For metals, when $N < 15$, $K=0$; when $16 < N < 75$, $K=N-15$; when $76 < N < 1000$, $K=0.8N$; and when $N > 1001$, $K=N-200$. As for J, when $N < 10$, $J=0$, and when $N > 10$, $J=10$ [34].

At specific moments throughout a test, the historic index is a measure of how severe the recent emissions have been compared to all emissions (with $HI=1$ representing average recent signal strengths), and severity is a measure of the strongest emissions received up to that moment. While the historic index may fluctuate up and down based on recent activity, the severity can only remain constant or increase.

IA has been used as an indication of component damage in a variety of tests [14, 17, 34, 19]. Once the HI and the S_r have been calculated for various stages during a test, they may be plotted against each other on an intensity chart. For a specific material, zones on this chart can relate to different stages of damage. A rapid increase in the severity can often be associated with the onset of structural damage [17]. Figure 2-7 shows an example of what a typical intensity chart looks like.

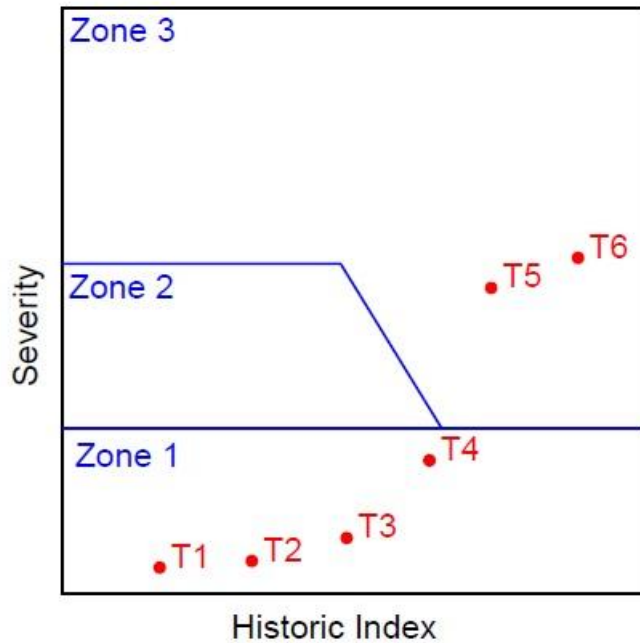


Figure 2-7: Intensity Chart Example

2.4.4 Cluster Analysis

Data clustering is a signal processing method that groups together signals with parameters within a specified range. In AE research, it has been used effectively to discern different event mechanisms from recorded activity [11, 16, 17, 25]. Cluster analysis is a more refined version of parameter analysis that is capable of using multiple parameters simultaneously to classify types of events. Figure 2-8 shows a theoretical example of data clusters in the relationship between each hit's amplitude and duration. As explained in section 2.4.1, this analysis may be able to separate out unwanted signals in post-processing.

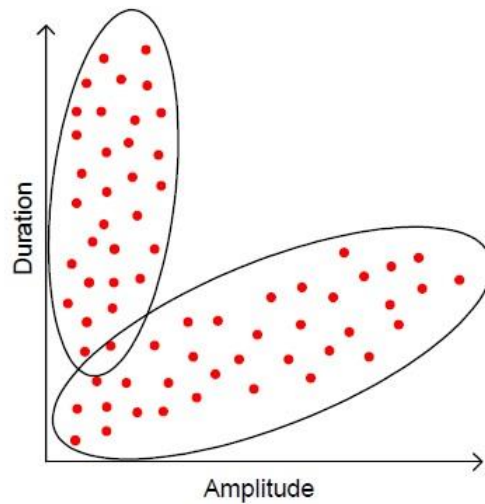


Figure 2-8: Example of Data Clusters

A similar analysis called k-nearest neighbor (k-NN) assigns predetermined cluster centers and uses an algorithm to compute the correct cluster for each event [16]. If a specific damage mechanism is known to cause AE with a certain range of parameters, then k-NN clustering can be used to ascertain which events correspond to that mechanism.

A more fundamental form of k-NN clustering, called k-means clustering, determines the locations of the cluster centers themselves through a complex algorithm that involves iterating multiple times to find the most effective arrangement of clusters [16]. When unsupervised pattern recognition is needed for AE data from new research, k-means provides a method of self-classification that may lead to useful data partitions [1]. Figure 2-9 shows the same data set as Figure 2-8 with their corresponding cluster centers, which can be determined through k-means clustering or assigned by k-NN clustering.

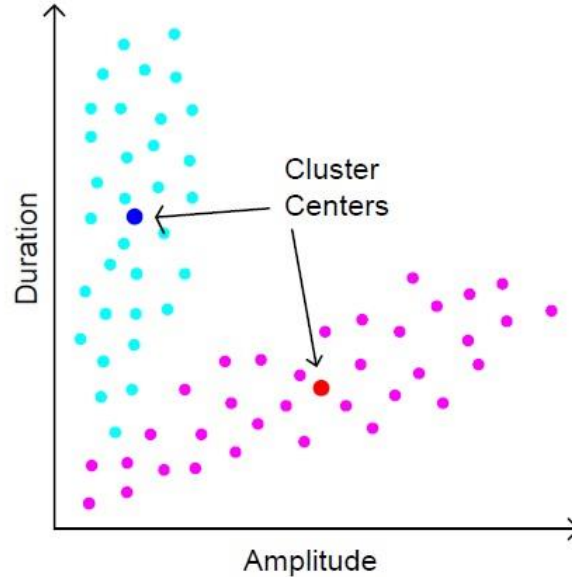


Figure 2-9: K-NN and K-Means Cluster Centers

This type of analysis is not limited to separating two parameters into two clusters. Any number of parameters can be partitioned into any number of clusters through the k-means and k-NN methods. In these more complicated analysis, each acoustic hit is assigned a vector of parameter coordinates such as duration, amplitude, rise time, energy, frequency centroid, and any others that may be helpful for classification. Typically these parameters are normalized so that the smallest and largest among the data set are equal to 0 and 1 respectively. The cluster centers are also represented by a certain coordinate's vector in this multi-parameter space. Hits are assigned to cluster centers by computing the shortest Euclidian distance between the hit's coordinates and the cluster center's coordinates.

CHAPTER III

EXPERIMENTAL METHODS

3.1 ACOUSTIC EMISSION MEASUREMENT

The following pages contain information regarding the methods of AE detection and analysis used in this study. Figure 3-1 shows the AE monitoring system used, including the transducers, data acquisition devices, and laptop display with monitoring software.

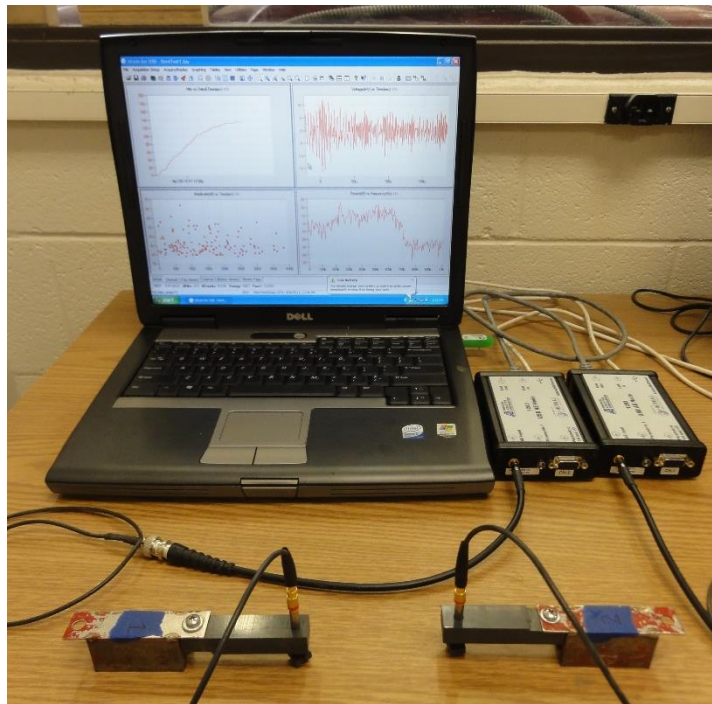


Figure 3-1: AE Monitoring System

The transducers used are two VP-3 Pinducers from the CTS Valpey Corporation. They have a defined frequency range of 1kHz to 10Mhz and a crystal diameter of 0.11". A magnetic mount was fabricated to securely attach the transducers to the test material. The couplant used is X30S Industrial Ultrasound Couplant from Staveley NDT Technologies Inc. Figure 3-2 shows the transducers inside and outside of the magnetic mount along with the couplant.



Figure 3-2: Transducers, Magnetic Mount, and Couplant

The data acquisition devices used are two MISTRAS 1283 USB AE Nodes. They have built in preamplifiers, a sampling frequency of 20MHz, and a gain of 40dB. The two nodes are connected to each other with an Ethernet cable for time synchronization.



Figure 3-3: MISTRAS Data Acquisition Device

The DAQs are wired into a laptop running AEWin™ Software which allows for live AE monitoring, waveform extraction, and other data processing capabilities. It can be customized to display a variety of graphed characteristics during and after testing. Figure 3-4 shows a typical software setup which displays—from the top left clockwise—a cumulative hits curve, the most recent waveform, its power spectrum, and a scatter plot of event amplitudes.

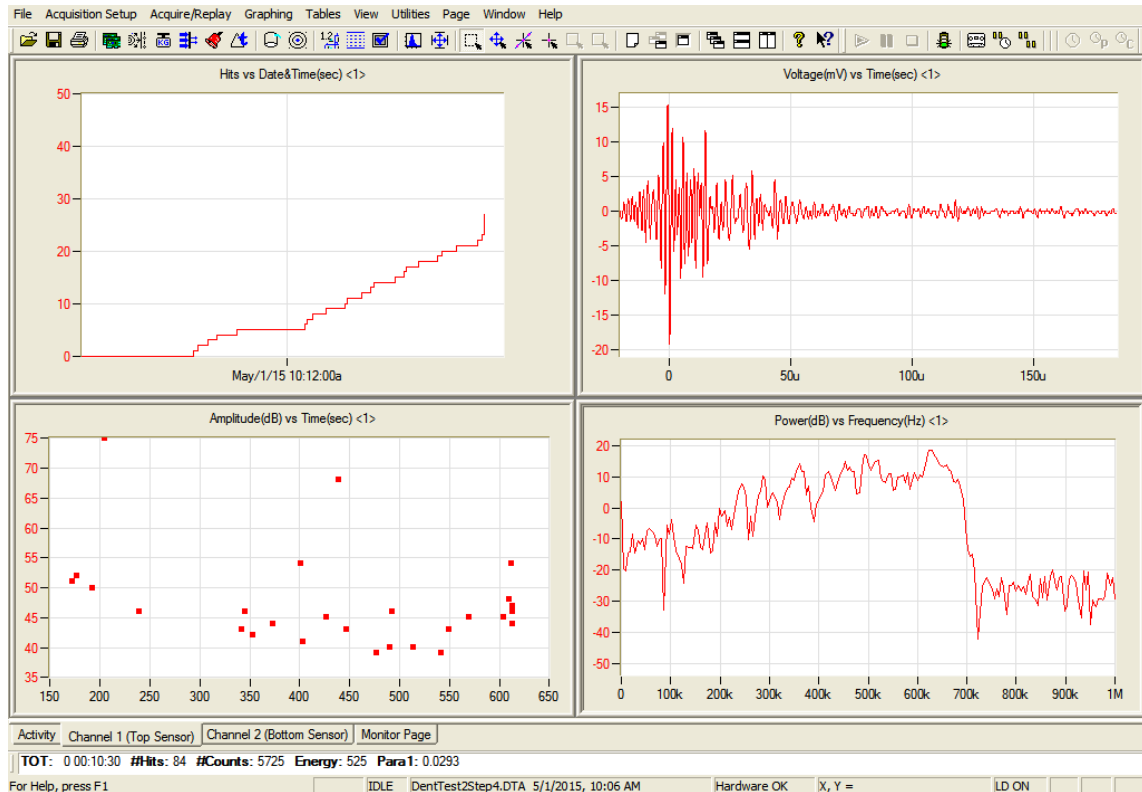


Figure 3-4: AEWin™ Software Screenshot

The AEwin™ software is set with a peak definition time (PDT) of 200μs, a hit definition time (HDT) of 800μs, and a hit lockout time (HLT) of 1000μs. These parameters help ensure that hits are defined correctly: see section 2.2.1 for their full definitions. The software's detection threshold can be adjusted before recording begins. For the tests in this study, the threshold was usually set slightly above the observed ambient background noise detected by the sensors. The software has a pre-trigger of 25μs which defines the beginning of each waveform a little earlier than the first count.

The software also has an adjustable analog filter for each sensor, which can be set with a lower bound of 1kHz or 20kHz and an upper bound of 200kHz, 300kHz, 400kHz, 500kHz, or 1MHz. It includes four additional specific ranges as well: 100kHz

to 300kHz, 400kHz, or 600kHz; and 400kHz to 990kHz. These analog filter limits are not precise: excitation detection continues beyond the limits and gradually drops to zero. The differences in these filters are explored in section 4.1.2.

3.1.1 Pencil Lead Break Tests

In order to test and calibrate the transducers prior to use, pencil lead break (PLB) tests—also known as Hsu-Neilson source tests—were performed on the test material [35]. These tests involved pressing down and snapping off a piece of mechanical pencil graphite to produce a relatively repeatable AE event. The graphite very slightly deforms the test material, and when it snaps off, the material rebounds and an elastic wave propagates through it. Slight differences in free lead length and contact angle can result in different event magnitudes in the test material [36]. However, PLB tests are still a useful way to test and calibrate AE transducers. For this research, a Papermate Precision 0.5mm mechanical pencil was used with 2H pencil lead. The free lead lengths were 3mm +/- 0.5mm, and the contact angles were 30° +/- 10°.

3.1.2 Post-Processing

All post-processing was carried out in MATLAB[®]. The AEwin[™] software is able to save individual waveforms as text files which include their detection time and voltage vector. MATLAB[®] scripts and functions were written to upload and analyze all of the

waveforms for any of the tests in this study. This allowed for a more robust analysis of the waveforms and their properties, a more selective data filtering, and additional post-processing tasks including the characterization techniques outlined in section 2.4. Additionally, certain parameters such as threshold had to be adjusted during post-processing in order to properly categorize each waveform's characteristics. All MATLAB[®] codes written for this paper are included in Appendix A.

Post processing allowed the AE data to be synchronized with the other instrumentation data. While all other instruments (strain gauges, LVDTs, load cells) were set to store their value in a spreadsheet periodically, the AE data was stored as individual waveforms captured at specific times. Codes were written to loop through each set of instrumentation data and to find their values at the detection time of each hit, as well as specific AE information at each instrumentation time stamp.

Even though a fixed threshold was used in AEwin[™] to detect and store the waveforms, post-processing in MATLAB[®] was done using a floating threshold to prevent misinterpretation of some of the larger waveforms. This floating threshold was the greater of 5% peak waveform amplitude and 0.01V (or 40dB). Figure 3-5 shows a high energy event from a pipe dent test which—when magnified—illustrates the need for this post-processing adjustment. Using the threshold set during the data acquisition, this event would have an inflated duration and could be misinterpreted during analysis.

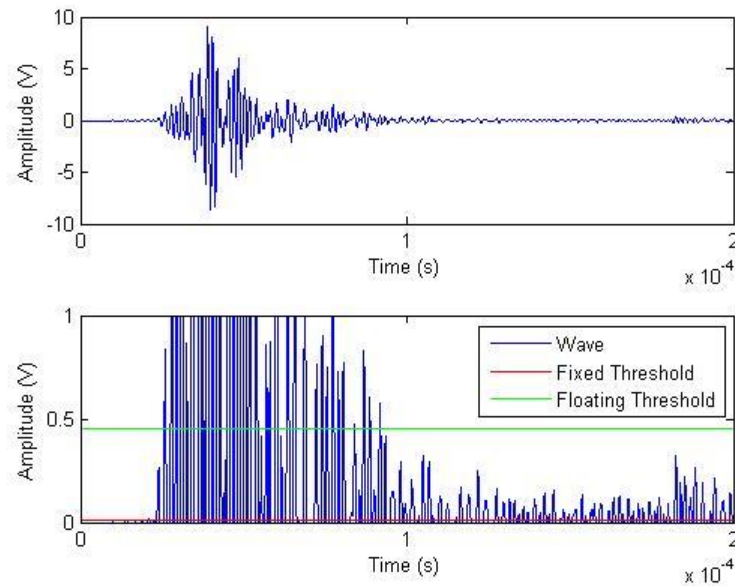


Figure 3-5: Justification of Post-Processing Floating Thresholds

While AEwin™ already uses a hit definition time (HDT) to define the peak of each waveform, sometimes waveforms include small blips or spikes that do not appear to be part of the original waveform. These are likely due to some sort of wave reflection or a momentary spike in background noise. Figure 3-6 shows a PLB waveform that illustrates the need for an additional HDT in post-processing. Without it, post-processing would assign an inflated duration to this waveform. An HDT of 25μs was chosen to properly categorize hits like this without cutting other waveforms off too early.

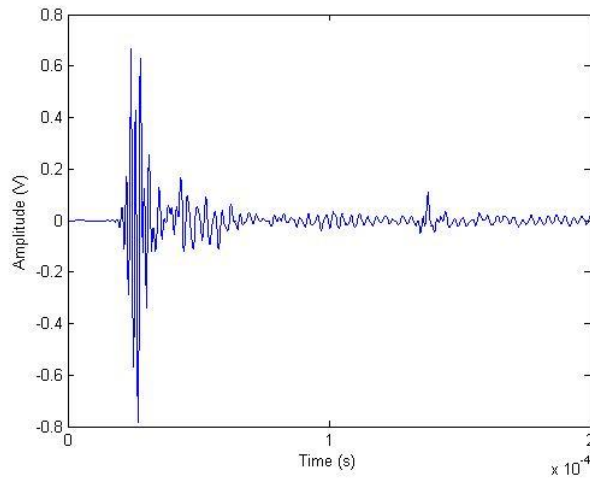


Figure 3-6: Justification of Additional Post-Processing HDT

On rare occasions, the Fourier transform used during post-processing revealed some incorrect power spectrum behavior which skewed the calculation of the frequency centroids. Figure 3-7 shows a pipe dent test waveform obtained with the 400kHz to 990kHz analog filter which resulted in an incorrect power spectrum. It is unclear what is causing this behavior, but it does not appear to have a major effect on the shape of the waveform. Post-processing was programmed to zero the power spectrum values at frequencies more than 100kHz outside of the analog filter.

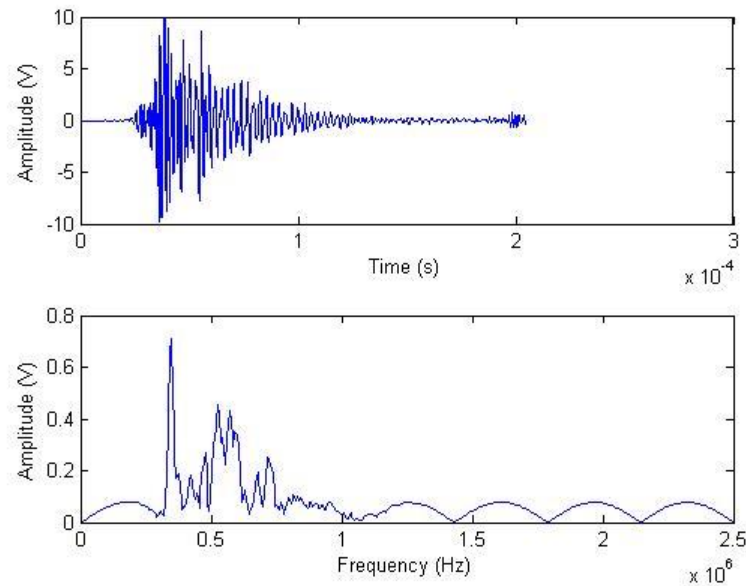


Figure 3-7: Justification of Post-Processing Power Spectrum Limits

Additional filters were applied in post-processing in order to remove unwanted signals. Unless otherwise stated, all hits with fewer than 3 counts above 0.01mV were discarded as they either represented random spikes in background noise or events generated far away from the sensors. This filter often removed a large portion of the recorded activity during the pipe tests, but the hits it removes contain little information and are unnecessary for analysis. Hits with frequency centroids below 200kHz were also removed as they were also found to be associated with background noise.

For pipe tests when sensors were placed on opposite faces of the pipe, a post-processing algorithm was used to detect signals that were picked up by both sensors but seemed to be the result of a single event. While the software has a HLT to prevent detection of reflections it does not apply to separate sensors. Signals registered by one sensor within 3ms after the other sensor were separated from the data file.

In order to better observe trends within the data, a function was written to sort events with specific ranges of parameters into groups. For example, this allowed for the individual analysis of only the hits with large amplitudes and short rise times. Since these groups of waveforms were a fraction the size of the entire data set, direct observation of individual waveforms was more feasible because a loop could graph the waveforms of all events within a specific range of parameters.

3.1.3 Transducer Mounts

Magnetic clamps—shown in Figure 3-8—were fabricated to mount the transducers onto the pipes. A small spring inside the plastic grip keeps the transducer pressed against the pipe surface. These clamps seemed to work well during the burst test and Dent Test 1 (see sections 3.4.1 and 4.4.1). However the transducers failed to record significant acoustic activity during the Dent Test 2 (see sections 3.4.2 and 4.4.2) even though a crack formed and the pipe eventually ripped open. PLB tests were conducted at various stages during each test to ensure that the sensors were still functioning, but the quality of the data acquired still turned out to be very poor. Tests were carried out to ascertain the reasons for the failure of the sensors, and these tests are detailed in section 4.1.

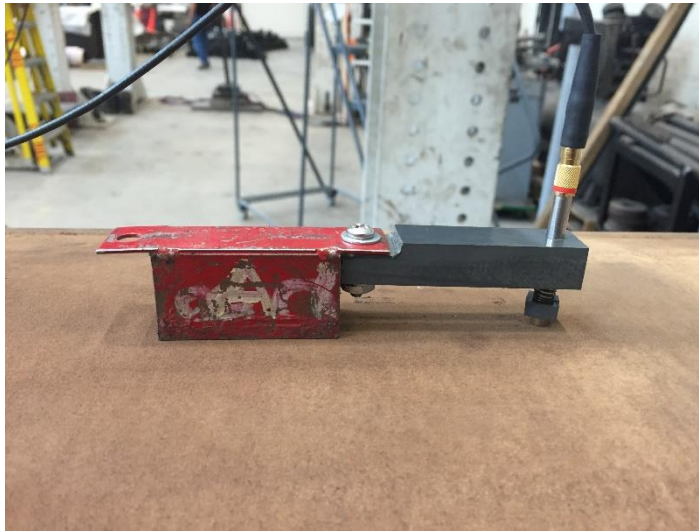


Figure 3-8: Original Transducer Mount on Dent Pipe

It was later found that the low activity of Dent Pipe 2 may be due to the material itself (see section 4.2), but the tests in section 4.1 revealed that sensor-to-source distance as well as sensor-to-source contact have significant impacts on the quality and consistency of the measured waveforms. This highlighted the inadequacies of the original magnetic clamp. The rigidity of the clamp sometimes forced the sensor tip to rest at tangential contact with the pipe surface, as shown in Figure 3-9.

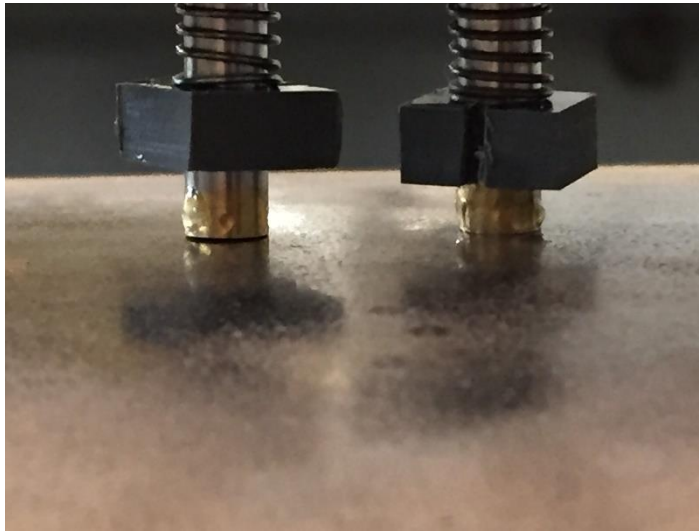


Figure 3-9: Example of Improper Sensor-to-Surface Contact

This potentially poor contact was exacerbated as the pipe dents grew and the curvature of the surface changed. The sensor recording the dent area was initially placed one foot away from the dent apex in order to avoid this rapid change in curvature. However, the tests in section 4.1.4 reveal that this distance results in significant attenuation of waveforms generated at the dent apex.

In order to more properly measure waveforms generated at the dent apex, a new type of transducer mount—shown in Figure 3-10—was fabricated. Unlike the magnetic clamps, this mount allows the sensor tip to remain flush with the pipe surface no matter what angle the magnets are at. The elastic bands keep the sensor pressed against the surface while allowing it to translate when the curvature of the pipe underneath the mount changes. This not only maintains a higher quality connection between the sensor and the surface, but allows the sensor to be placed much closer to the dent apex. This

mount allowed the transducer to successfully record acoustic activity throughout Dent Test 3 (see sections 3.4.3 and 4.4.3).

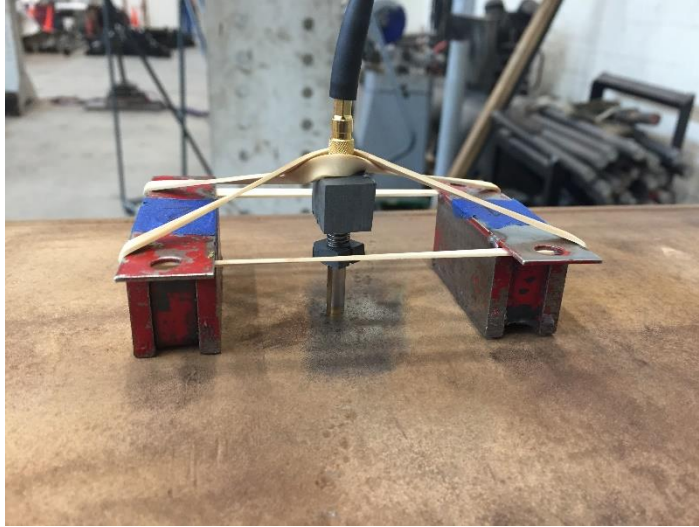


Figure 3-10: New Transducer Mount

3.2 TENSILE TESTS

Tensile tests performed for this study were done in an MTS Tensile Testing Machine, shown in Figure 3-11. Tests were conducted with API X52 specimen made from the same steel as Dent Pipe 2 (see section 3.4) as well as another spare set of tensile specimen leftover from a previous study. All specimen were 4" long with a gauge length of 1.5", a diameter of 0.25", and a grip diameter of 0.33". Table 3-1 shows the primary chemical composition (by weight) of the two types of specimen as determined by optical emission spectrometer.

	C	Mn	P	S	Si	Ni	Mo	Cr	Cu
API X52	.26	.91	.018	.027	.01	.08	.02	.05	.07
Spare Steel	.21	.61	.013	.022	.09	.02	<.01	.04	.02

Table 3-1: Tensile Test Chemical Analysis

The tests were conducted in an MTS Tensile Testing Machine, shown in Figure 3-11. An extensometer was used to accurately measure strains. A small patch was shaved down on each specimen's grip surface so that the transducer could make flush contact with the specimen and more accurately measure its emissions. The transducer was held in contact with the specimen using elastic bands and plastic grip with an internal spring, so that the sensor kept good connectivity throughout the tests. Figure 3-12 shows a tensile rod gripped in the machine along with the transducer and the extensometer.



Figure 3-11: MTS Tensile Testing Machine

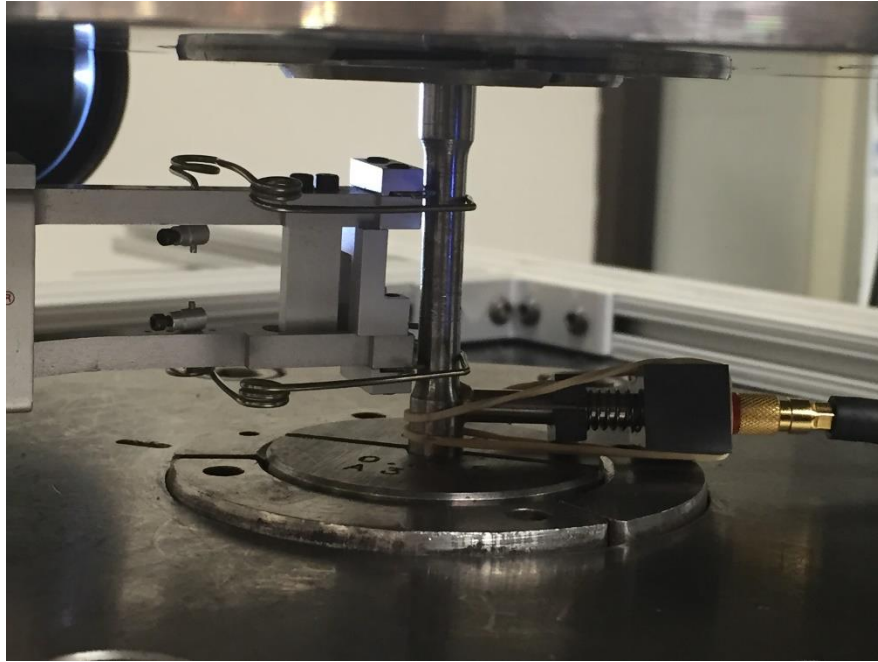


Figure 3-12: Tensile Test Sensors

Tests were conducted with the 100-600kHz and the 400-990kHz analog filters in order to observe the recorded emissions in both spectrum ranges. A threshold of 40dB was applied to remove unwanted noise generated by the machine during testing.

3.3 PIPE BURST TEST

A single pipe burst test was carried out during the course of this research. In this test, a decommissioned gas pipeline segment with known internal material defects was pressurized with water until the pipe burst. Caps were welded onto the ends of the segment. There were two defects located at opposite sides of the midsection. The pressurization proceeded in pre-determined steps, and at one of those steps the pipe pressure was reduced so that it could be inspected safely. Figure 3-13 shows the burst test pipe with its pressurization pump, and Figure 3-14 shows the burst test pipe once it was underneath protective shielding cut from an old dent test pipe.



Figure 3-13: Burst Test – Pipe with Pressurization Setup



Figure 3-14: Burst Test – Pipe with Protective Shielding

Since this test was carried out a few days after the AE monitoring hardware and software was acquired, the test served as a convenient trial run for the new AE system. The DAQs and all wire connections were wrapped in tarp in case the pipe ruptured and the test setup was flooded with water. A transducer was placed 4" away from each defect in an attempt to determine which defect would be the one to rupture. Figure 3-15 shows the transducer mounted next to the location of one of the defects.

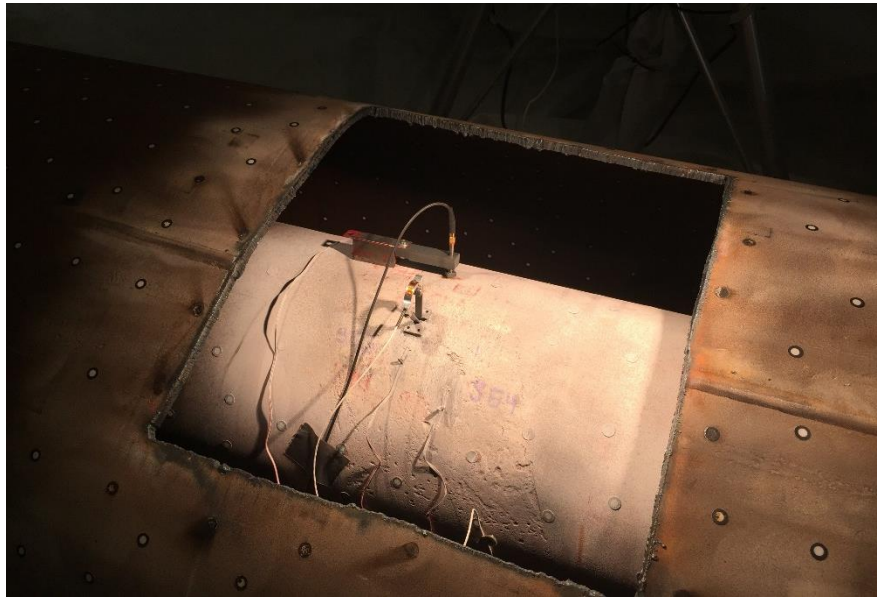


Figure 3-15: Burst Test – Transducer Next to Defect

The sensors were set to the 100-600kHz analog filter. The threshold of detection was initially set to 30db, which was slightly above the observed ambient noise level. However once the pressurization began, the sensors began registering hits at an alarming rate (around 5000 per minute) and it seemed that the threshold must be way too low. The AE system was paused and the threshold was reset to 40db.

While the pressure was relieved and the test was paused, one of the AE sensors was removed. The strain gauges had already indicated which defect was going to rupture. The sensor monitoring the other defect was removed because of concerns that failure of the pipe might damage both sensors, preventing AE monitoring of the pipe dent test scheduled for the following week. The pipe did burst and the remaining sensor survived the rupture. Figure 3-16 shows the burst test pipe after the rupture.

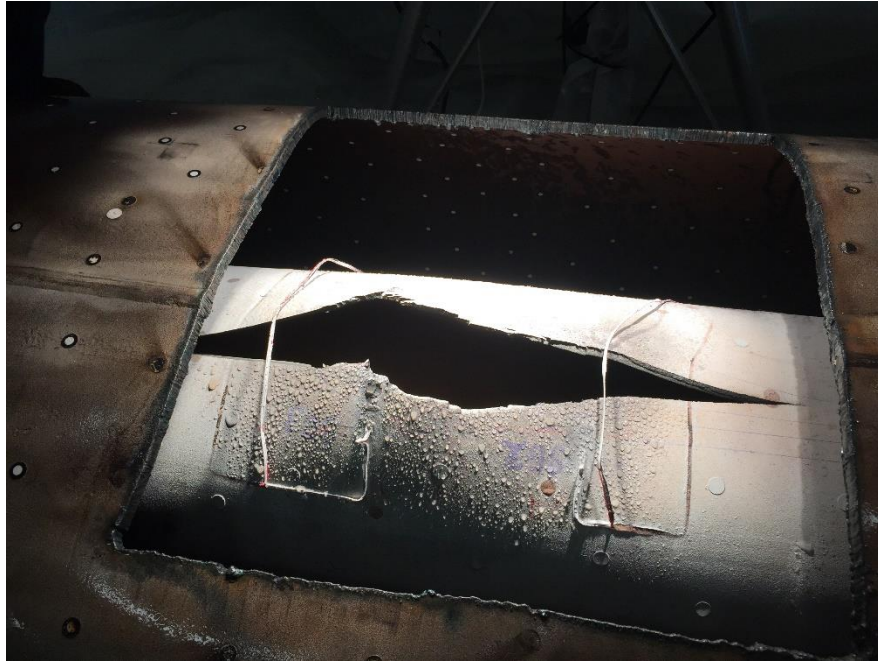


Figure 3-16: Burst Test – Post Failure

3.4 PIPE DENT TESTS

Three pipe dent tests were carried out during the course of this research. The pipes were decommissioned gas pipeline segments with 3/8ths inch thick walls and additional properties shown in Table 3-2. All pipes had a 6" strip of additional pipe metal welded around the ends to simulate an adjacent attached pipe segment and act as stiffness reinforcement. The pipes rested on a wooden saddle and were dented with a 200 kip actuator supported by the frame shown in Figure 3-17.

	Steel Grade	Diameter	Length	Dent Location
Dent Pipe 1	API X65	36"	133"	Pipe body
Dent Pipe 2	API X52	34"	124"	Seam weld
Dent Pipe 3	API X65	36"	114"	Girth weld

Table 3-2: Dent Test Physical Properties

Table 3-3 shows the primary chemical composition (by weight) of the two types of pipe steels as determined by optical emission spectrometer.

	C	Mn	P	S	Si	Ni	Mo	Cr	Cu
API X65	.12	1.46	.011	<.002	.23	.02	<.01	.05	.02
API X52	.26	.97	.013	.022	.02	.05	.01	.03	.04

Table 3-3: Dent Test Chemical Analysis

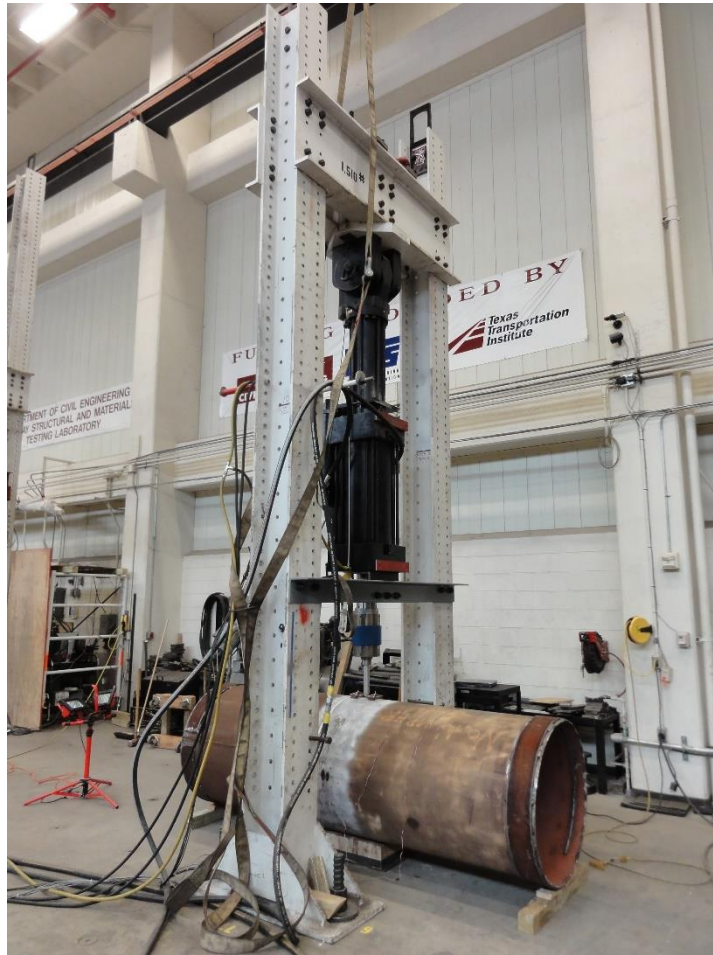


Figure 3-17: Denting Frame with Dent Pipe Underneath

Dent Pipes 1 and 2 had seam welds all the way across their length. Dent Pipe 3 had a girth weld directly in the middle, and two seam welds 160 degrees away from each other on either half. All pipes were dented half way between the ends. Dent Pipe 1 was dented 180 degrees away from its seam weld, Dent Pipe 2 was dented directly on its seam weld, and Dent Pipe 3 was dented on its girth weld in-between its seam welds.

The dents were created at a rate of 0.034 inches per minute. The indenter itself was a 1.5" diameter aluminum shaft with a semi-spherical end. A video camera was placed inside the pipe for live observation of any crack growth occurring in the dent

area. Dent depths were measured with an LVDT placed inside the pipes directly beneath the dent apex. The denting was paused every 60 minutes or so in order to reset the LVDT, which had a stroke length of 2". Additionally, the tests were paused at specific intervals so that a laser scanning system could be used to accurately measure strains throughout the pipe. If the tests lasted longer than one day, the loads were maintained overnight and the denting was resumed in the morning.

Four strain gauges were used for each test: two in the longitudinal axis of the dent apex, and two in the circumferential (or hoop) axis of the dent apex. For Dent Tests 1 and 3 which used a 36" diameter pipe, the gauges were placed 14" and 28" away from the dent apex in each direction. For Dent Test 2 which used a 34" diameter pipe, the gauges were placed 13" and 26" away. The two closer gauges have been labeled SGL1 (longitudinal) and SGH1 (hoop) and the two farther strain gauges have been labeled SGL2 and SGH2.

3.4.1 Dent Test 1

Figure 3-18 shows Dent Pipe 1 underneath the denting frame. The dent was pushed up to 11" without the appearance of a crack, and the test was halted. Figure 3-19 shows the final deformation of the dent through the pipe.



Figure 3-18: Dent Test 1 – Pre Dent, Pipe underneath Denting Frame



Figure 3-19: Dent Test 1 – View of Final Dent through Pipe

Like the burst test described in section 3.3, Dent Test 1 was scheduled very soon after the acquisition of the AE hardware and software. Knowledge of the relationships between event source, acoustic waveform, and transducer signal quality were still limited.

For this test, one transducer was placed on top of the pipe 12" away from the dent apex, and the other transducer was placed underneath the pipe right next to the seam weld and 12" away from the center on the opposite side. These locations were chosen in an attempt to determine whether certain acoustic emissions were created by the dent area or by the support reaction. The sensor on top was placed 12" away from the dent because the curvature of the pipe around the dent tends to warp significantly, and a 12" distance kept it away from the increasing curvature. The sensor on the bottom was also placed 12" off center to match. The sensors were set with an analog filter of 100-600kHz and a threshold of 40db, which was above the ambient noise recorded by the sensors while attached to the pipe.

3.4.2 Dent Test 2

Figure 3-20 shows Dent Pipe 2 is shown underneath the denting frame. When the LVDT was being reset after a dent depth of 6% diameter, a crack was noticed directly underneath the dent apex, shown in Figure 3-21. The dent was increased until a sudden expansion of the crack pushed the indenter through the pipe, resulting in the deformation shown in Figure 3-22.



Figure 3-20: Dent Test 2 – Pre Dent, Pipe underneath Denting Frame

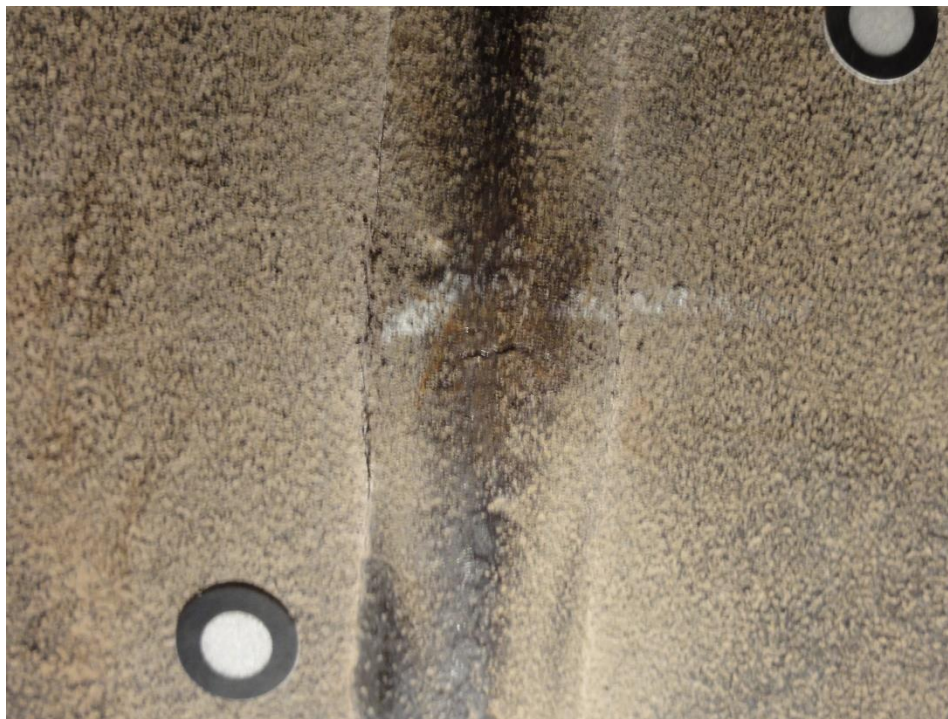


Figure 3-21: Dent Test 2 – Initial Crack inside Pipe at Dent Apex



Figure 3-22: Dent Test 2 – Final Crack inside Pipe at Dent Apex

For Dent Pipe 2, the two transducers were placed in the same relative locations using the same analog filter and threshold as Dent Pipe 1. This was because the acoustics proved successful in recording the activity in Dent Pipe 1 even though there was no crack formation. However, during Dent Test 2 before the crack was noticed, barely any signals were being registered by the sensor on top of the pipe. This was concerning, and in an attempt to better capture the crack growth events, the top transducer was moved to 3" away from the dent apex until the crack broke through. Unfortunately this only helped for a while and the sensor seemingly failed to record substantial activity during the final stages of crack growth.

3.4.3 Dent Test 3

While Dent Tests 1 and 2 were closely scheduled, there was time before Dent Test 3 to figure out why the sensors failed to function properly during Dent Test 2. As explained in sections 3.1.3 and 4.1, it may have been due to poor sensor-to-surface connectivity and improper sensor-to-source distance. Additionally, the results in section 4.2 reveal that the low activity may have also been due to the API X52 of Dent Pipe 2.

Figure 3-23 shows Dent Pipe 3 underneath the denting frame. By a dent depth of 3% diameter, a crack was beginning to form at the edge of the interior girth weld, shown in Figure 3-24. By about 6% another crack was forming on the other edge of the weld. This crack gradually grew until the test was stopped at a depth of 12%, and the final crack is shown in Figure 3-25.



Figure 3-23: Dent Test 3 – Pre Dent, Pipe underneath Denting Frame



Figure 3-24: Dent Test 3 – Initial Crack inside Pipe at Dent Apex



Figure 3-25: Dent Test 3 – Final Crack inside Pipe at Dent Apex

The new transducer mounts described in section 3.1.3 were used for this dent test, allowing the sensor to be placed much closer to the dent apex than in previous tests. It was decided to place both sensors on top of the pipe and set them to different frequency filters in case that was one of the reasons the acoustics failed in Dent Test 2. As shown in Figure 3-26, the sensors were mounts were placed on opposite sides of the dent apex in the longitudinal direction. The sensor tips were set at 3" away from the dent apex in either direction. Sensor 1 (on the right) kept the 100-600kHz analog filter while sensor 2 (on the left) was set at the 400-990kHz filter. Further discussion of these analog filters can be found in section 4.1.2.

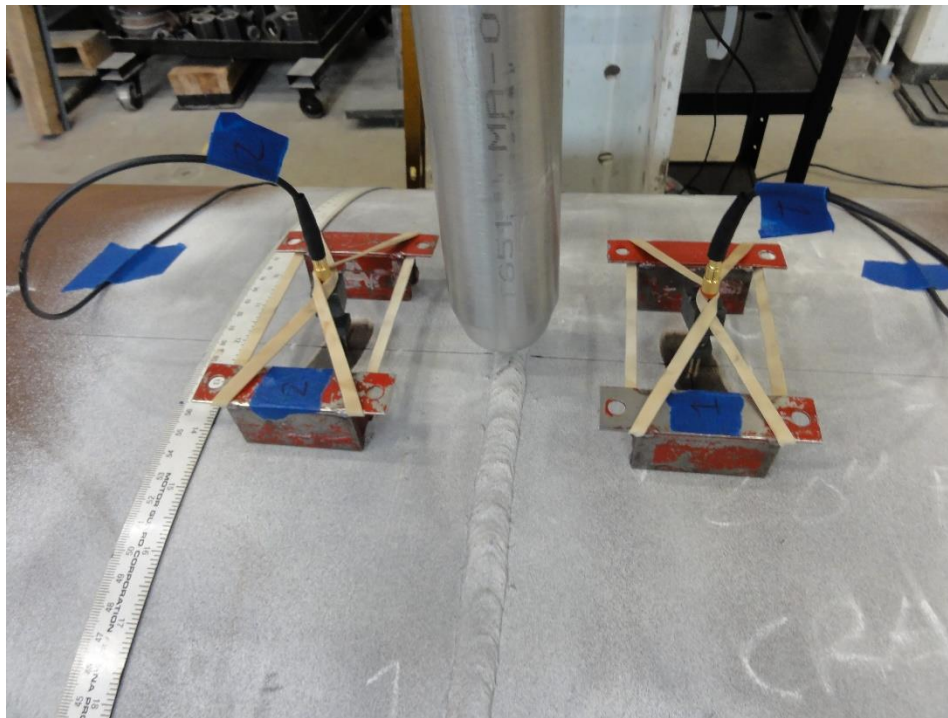


Figure 3-26: Dent Test 3 – Transducer Placement next to Indenter

While the sensors were being calibrated, sensor 1 was picking up a background noise between 45-48db. This was already a good sign that the sensor-to-surface connectivity was improved, since the background noise before previous tests was observed to be below 40db. Sensor 2 was not picking this up because the noise had a lower frequency content. Sensor 1 was set at a threshold of 50db to avoid this noise. Since there was very little noise at the frequencies measured by sensor 2, it was set at a low threshold of 35db (around 6mV). This threshold was set just in case the 40db cutoff was one of the reasons the acoustics failed in Dent Test 2.

The new sensor mounts worked very well at maintaining proper sensor-to-surface contact even as the curvature of the pipe around the dent increased. Figure 3-27 shows how the mounts had adjusted when the dent was at its final depth.

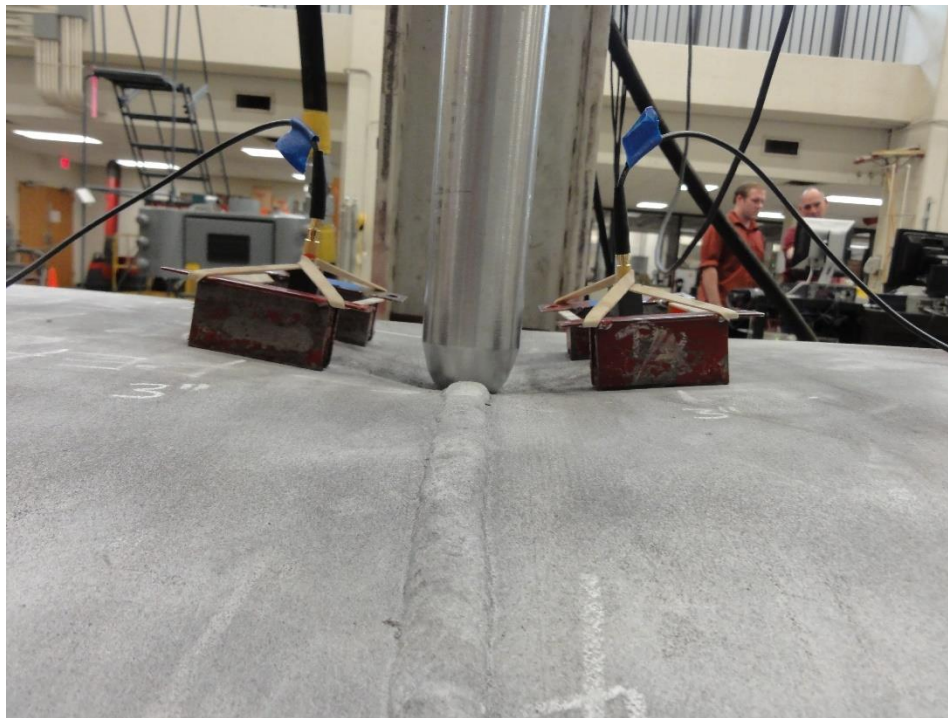


Figure 3-27: Dent Test 3 – Transducer Positions at Final Dent Depth

CHAPTER IV

EXPERIMENTAL RESULTS

4.1 TRANSDUCER TESTS

A number of tests were carried out to learn more about the characteristics of the transducers as well as the behavior of elastic waves inside the pipe. These tests involved creating PLB sources at specific locations along the pipe to better understand the relationship between acoustic events and the signal generated by the monitoring system. Studies have shown that although PLB sources are useful for their repeatability, variations in free lead length and contact angle can result in slightly different signals [36]. When possible for the following preliminary tests, multiple PLB sources were generated and averages were used for analysis. All of these tests were done using the original sensor mount described in section 3.1.3, and their results helped justify the need for a new sensor mount.

4.1.1 Sensor Calibration Tests

In order to determine how similar the two transducers used in this study are, they were placed two inches away from each other and PLB sources were applied half way between them. At first, the pipe surface was unpolished and this resulted in significantly different signals received by the two sensors. Additionally, the sensors' responses appear

to be somewhat sensitive to proper couplant application, perhaps due to the fact that the pipe's surface is slightly curved and the transducer cannot make fully flush contact.

Once the surface under the sensors was polished and proper couplant was applied, the sensors were producing relatively similar looking signals from the PLB sources, however their parameters were still different. Ten different PLB sources were stored, and their averages are shown in Table 4-1. The signals for one of these PLB sources is shown in Figure 4-1.

	Sensor 1	Sensor 2
Amplitude (V)	0.73	0.59
Frequency Centroid (kHz)	443	432
Duration (μ s)	36	47
Rise Time (μ s)	5.4	2.1
Counts	12	16
Signal Strength (Relative)	27.6	36.0

Table 4-1: Calibration Test 1 Parameters

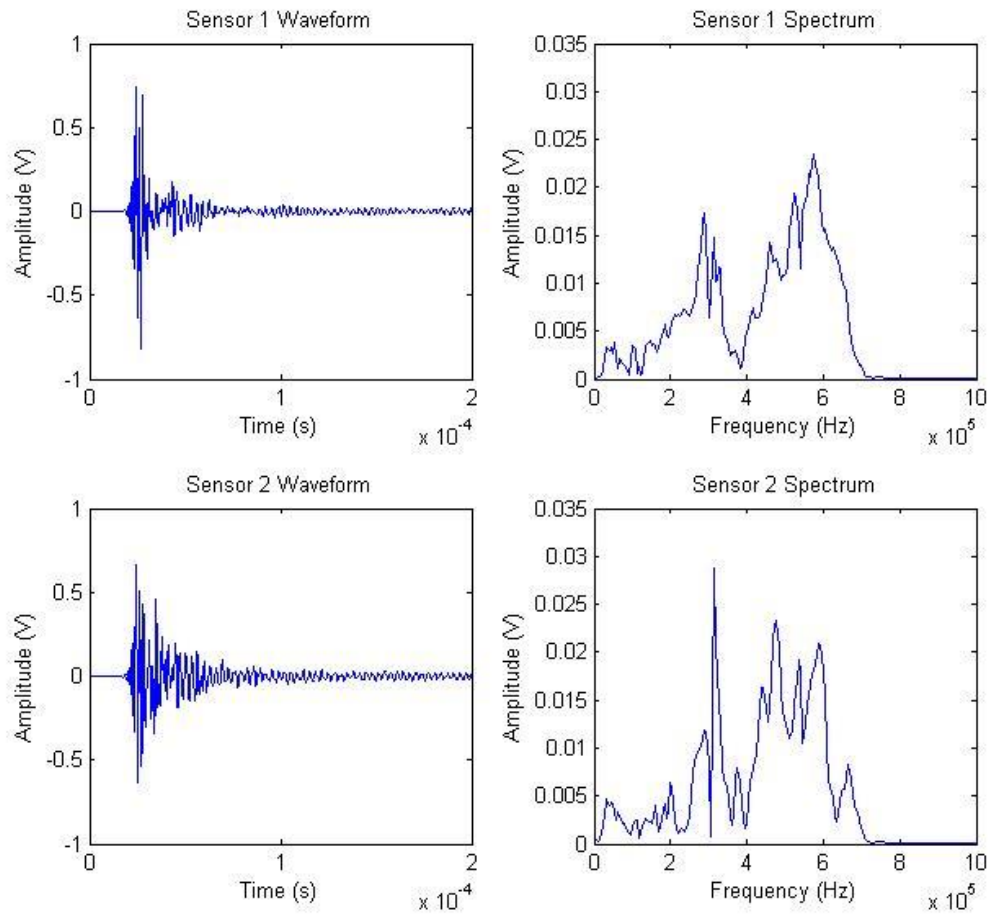


Figure 4-1: Calibration Test 1 Waveforms

The amplitudes and rise times for sensor 1 were always higher than sensor 2, but the opposite is true for durations and signal strength. Although the waveforms were generated by the same source at the same distance, and although the resulting signals look fairly similar in shape, something is causing their parameters to vary. This also effects the shapes of their power spectrums as shown in Figure 4-1, although their frequency centroids remained similar.

Figure 4-2 shows the first portion of both signals in Figure 4-1. For the first three oscillations, the signals are identical and remain synchronized. However something causes sensor 1's peak to continue to grow while sensor 2 starts to drop, resulting in the higher amplitude and the longer rise time. But sensor 2 remains at relatively larger values resulting in the longer duration and higher signal strength. It seems that the PLB source may be reflecting off of something and returning to the sensors. This could cause constructive and destructive interference at various stages of the waveforms, skewing the parameters shown in Table 4-1. It may be due to the pipe's seam weld, which happened to be 6" away from the test and closer to sensor 1 than sensor 2.

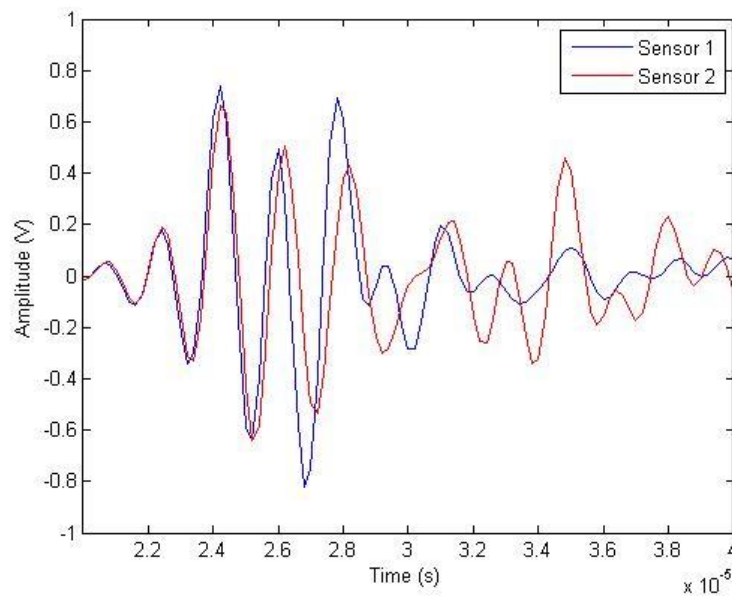


Figure 4-2: Calibration Test 1 Comparison

A similar calibration test was repeated with the transducer tips placed 1" away from each other (as close as their magnetic clamps would allow) and 10 PLB sources were generated directly in between the two sensors. The transducers were placed as far

away as possible from any geometric anisotropies, like the seam weld of the end of the pipe. This time the amplitudes were almost identical, however some of the other parameters remained oddly different. Table 4-2 shows the averages parameters of the 10 PLB waveforms.

	Sensor 1	Sensor 2
Amplitude (V)	2.95	3.03
Frequency Centroid (kHz)	424	438
Duration (μ s)	33	93
Rise Time (μ s)	2.2	1.9
Counts	11	27
Signal Strength (Relative)	86	153

Table 4-2: Calibration Test 2 Parameters

Figure 4-3 shows one of the PLB signals and power spectra for each sensor. Even though the signal magnitudes are identical, sensor 2 still shows an elevated ring-down as the wave dissipates. Both this figure and Figure 4-1 show that sensor 2 has a very sharp frequency spike at 300kHz that sensor 1 is missing. This may be linked to the differences in signal ring downs between sensor 1 and sensor 2. Either sensor 2 is registering something that it shouldn't or sensor 1 is not correctly acquiring the waveform's ring down.

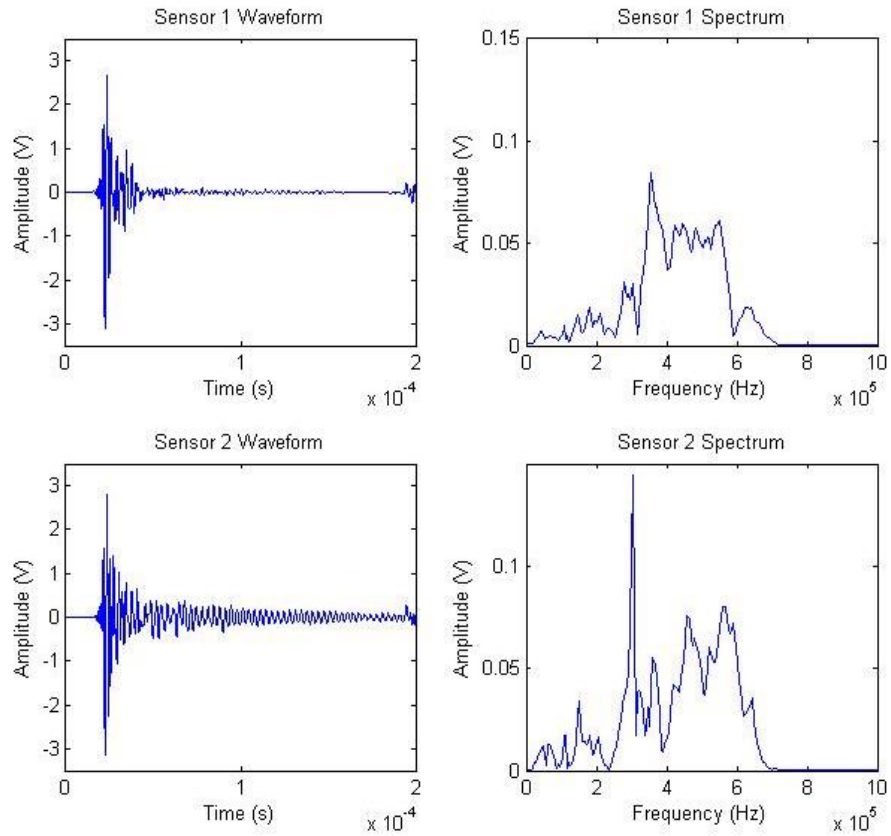


Figure 4-3: Calibration Test 2 Waveforms

Figure 4-4 shows the first portion of both signals from Figure 4-3. This time, unlike in the calibration test comparison shown in Figure 4-2, both signals remain identical for the primary portion of the elastic wave. However once the wave passes its maximum the two signals grow out of phase. About 40 μ s into the waveform, the signal from sensor 1 has practically diminished but the signal from sensor 2 is still ringing down.

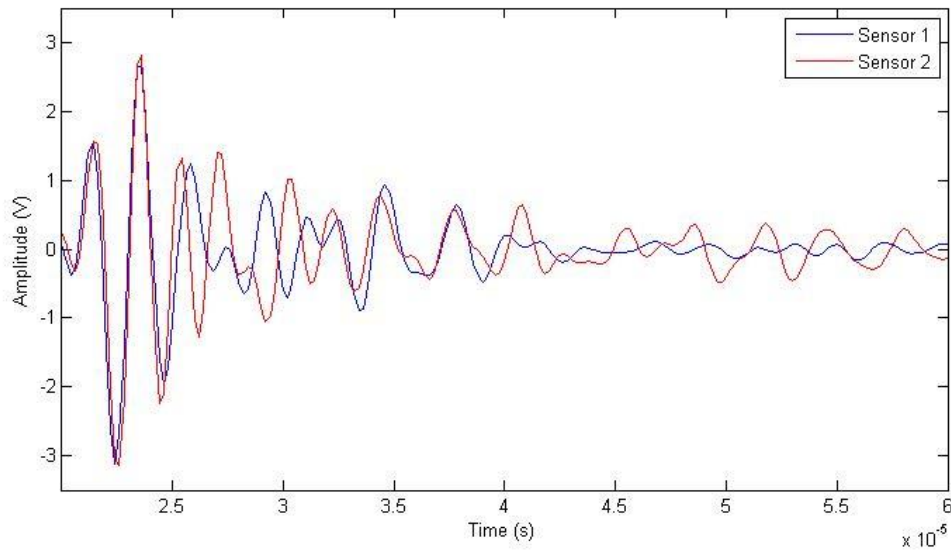


Figure 4-4: Calibration Test 2 Comparison

There are five potential causes of this discrepancy between sensors: positioning, sensor-surface contact, the DAQ/software, the sensors themselves, or simple errors in the PLB test locations. The results of the first calibration test suggest that positioning does have an effect on the waveforms. But the persistence of the discrepancy in the second calibration test—when the sensors were placed as close as possible, and far away from a weld or the end of the pipe—shows that positioning likely isn't the cause of the difference. When the same calibration test was carried out after switching the DAQs connecting each sensor to the computer, the same irregularity arose in the waveforms obtained by the same transducer, so it is not a result of signal processing.

Since the transducer crystals are flat and the pipe is curved, the irregularity may be a result of the quality of the surface/sensor connectivity. Previous PLB tests did show that waveform quality is affected if the pipe surface is not polished, or if couplant is not properly applied. The sensor calibration test was carried out again on a flat 16" by 9.5",

2" thick plate of steel to see how the transducers respond to waves on flat surfaces. Table 4-3 shows the averages parameters of the 10 PLB waveforms.

	Sensor 1	Sensor 2
Amplitude (V)	2.45	2.41
Frequency Centroid (kHz)	437	470
Duration (μ s)	41	45
Rise Time (μ s)	3.0	2.2
Counts	14	16
Signal Strength (Relative)	94	99

Table 4-3: Calibration Test 3 Parameters

The calibration is not perfect, but it is much closer than the previous tests. It should be noted that the small plate was causing reflections in the waveform, so a floating threshold of 10% maximum amplitude was used to more appropriately capture the duration and number of counts. This is why these values are smaller than previous tests. What is most important is that there is only a 5% difference in the signal strengths, which is the area underneath the rectified signals. Figure 4-5 shows one of the PLB signals and power spectra for each sensor, and Figure 4-6 shows the same signals overlapped for comparison.

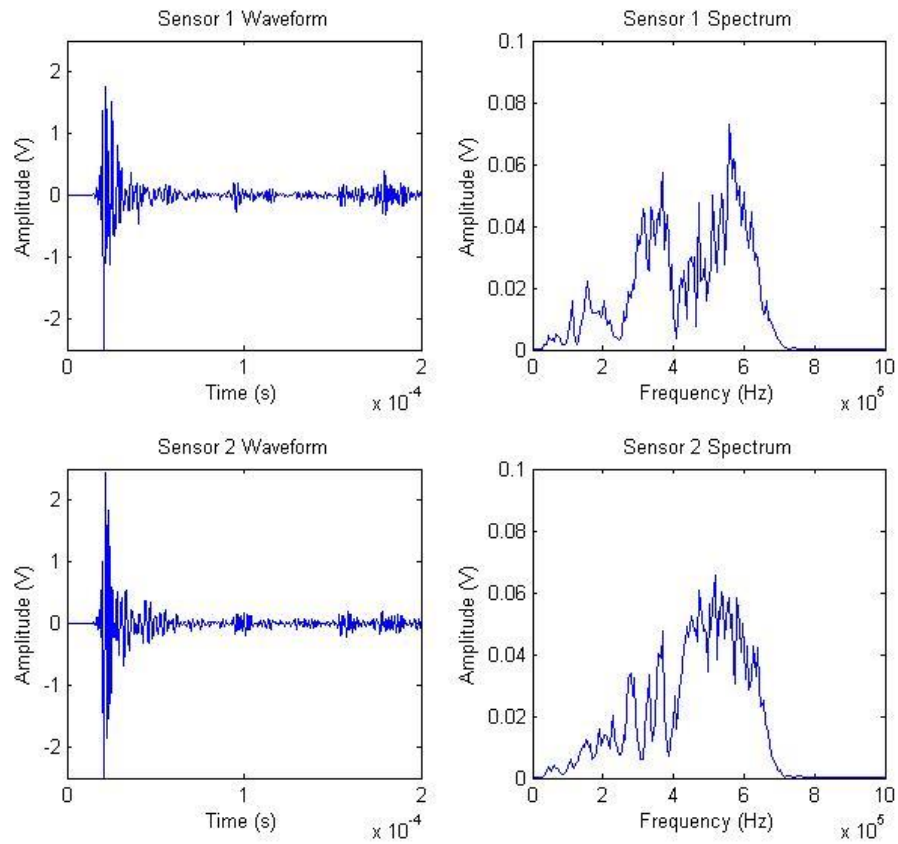


Figure 4-5: Calibration Test 3 Waveforms

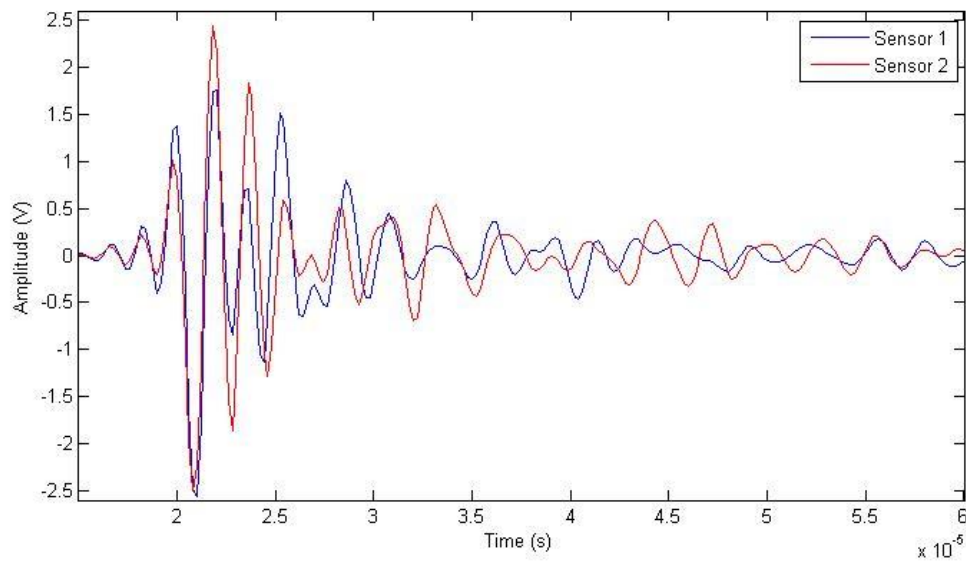


Figure 4-6: Calibration Test 3 Comparison

The remaining differences in waveform signals must be due to slight errors in the locations of the PLB tests and/or differences in the sensors themselves. Perhaps one of the two piezoelectric crystals was scratched or damaged in a prior test. Since sensor 2 was the one used for the whole burst test, it could have been damaged when the pipe failed.

These calibration tests show that there are only minor differences between the two sensors used in this research. However, they also suggest that the imperfect sensor-to-surface contact between the pipe and the sensor is likely to introduce errors and/or irregularities in the measured waveforms. This fact will have to be accounted for in the analyses that follow.

4.1.2 Software Parameters Tests

Since the AEwin[™] uses a variable detection threshold and analog filter, PLB tests were conducted with different software parameters to observe their effect on AE signals. Five PLB sources were created one inch away from a sensor with each of five different combinations of filters and thresholds, and their average results are shown in Table 4-4.

	Test 1	Test 2	Test 3	Test 4	Test 5
Filter (kHz)	100-600	100-600	20-10000	20-500	400-990
Threshold (dB)	45	60	45	45	45
Amplitude (V)	0.65	0.64	1.22	0.53	1.20
Frequency Centroid (kHz)	461	456	670	399	650
Duration (μ s)	44	27	50	79	42
Rise Time (μ s)	5.6	3.8	3.8	7.7	3.9
Counts	16	10	23	19	18
Signal Strength (Relative)	25	23	36	27	31

Table 4-4: Software Parameter Tests

When the system threshold is increased, waveforms have the same amplitude, frequency, and signal strength. But their duration and rise time are shorter, and they have fewer counts, which is to be expected. The AEwin[™] threshold determines which waveforms are stored and which are not. But the threshold dependent parameters are calculated in post-processing and the threshold can be changed in MATLAB[®]. In test 2, it was raised to illustrate this difference. As long as there is consistency between tests in post-processing, the threshold dependent parameters will be relatable to each other.

Figure 4-7 shows the power spectrum for a single signal from of the five tests. Test 4 gives the most complete spectrum of the frequencies excited by the PLB tests. These are the naturally excited frequencies of the metal, and should be similar to the frequencies excited by deformation events. The spectrums also illustrate the fact that these analog filters are not sharp cutoffs, rather frequency detection beyond their limits gradually drops off.

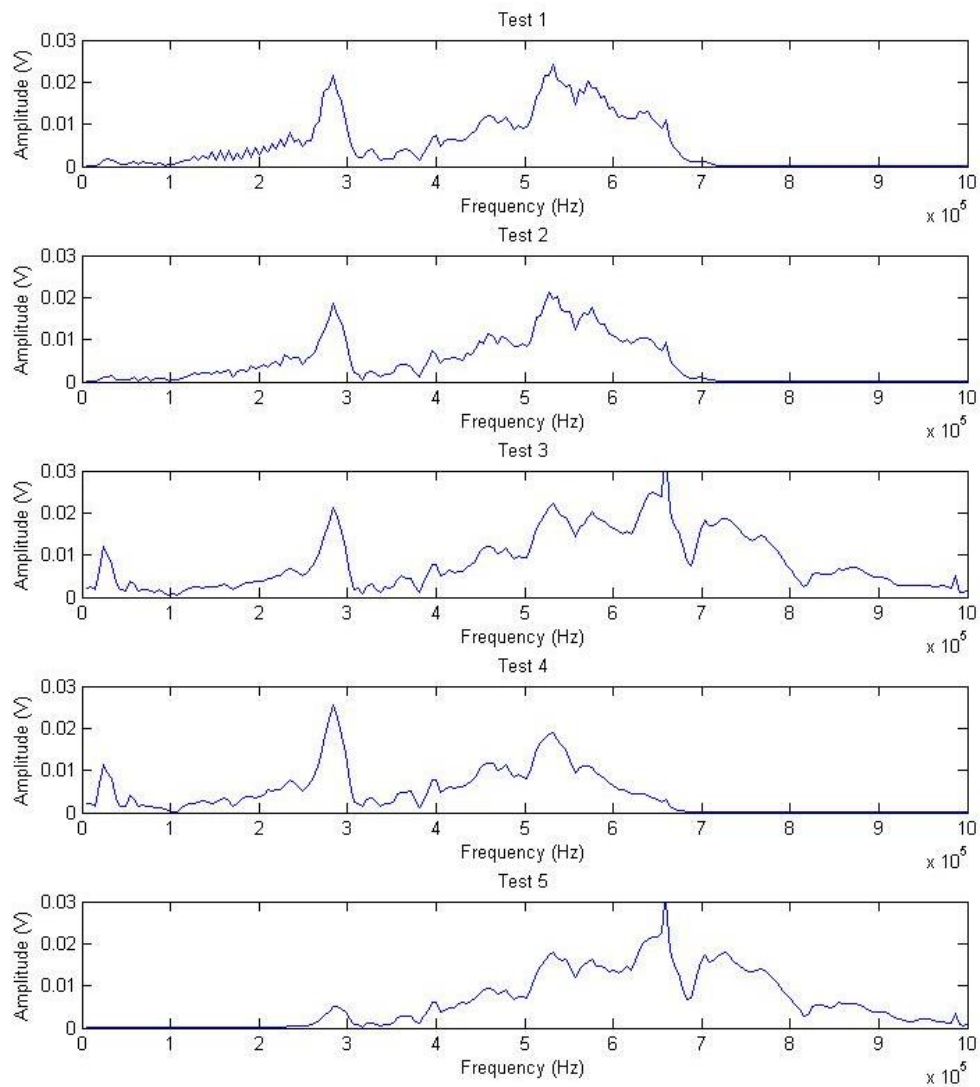


Figure 4-7: Analog Filter Frequency Spectra

The analog filter has a more noticeable effect on the signal's shape and parameters. The tests with higher frequency content have larger amplitudes, high frequency centroids, and greater signal strength. Tests 3 and 4, which allow slower frequencies to pass through, have the longest durations. The reason for this is illustrated in Figure 4-8, which is one of the waveforms from test 3. The low frequency spike

shown in the spectra for test 3 and 4 is manifested in the waveform as a slow, overarching waving pattern. It sometimes creates a hump halfway through the waveform which artificially inflates the duration by putting more of the waveform above the threshold.

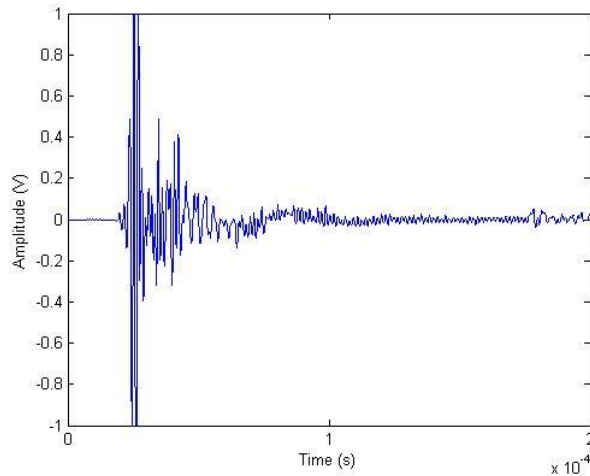


Figure 4-8: Waveform with Low Frequency Content

Unfortunately, there is no perfect analog filter in the AEwinTM software for the frequencies excited inside the pipe. The 20kHz-1MHz filter lets in too much background noise and distorts the signals slightly. The lower frequencies can be removed in post-processing, but since they are included in signal detection, they force the threshold to be raised to avoid recording their ambient noise. This might result in the software missing relatively quiet, high frequency emissions. The 100kHz-600kHz filter picks up the majority of the frequency content while ignoring the low frequency noise, but leaves out some of the high frequency content. And the 400kHz-990kHz filter keeps all of the high frequency content, but removes the spike shown around 280kHz. For most tests, the

100kHz-600kHz filter was used, as it is closest to representing the bulk of the frequencies excited in the steel.

4.1.3 Wave Speed Test

A test was carried out to estimate the speed that elastic waves travel through the pipe metal. The sensors were placed one foot apart, and ten PLB sources were created another foot away from sensor 2, so that the difference in detection time could be used to estimate the wave speed. Since the waves travel very quickly, small errors in the sensor placement distance will translate into larger errors in the wave velocity estimation. A caliper was used to mark the foot length to minimize error, however the sensors were still placed on the marks by sight, with an approximate error of $\pm 1/16^{\text{th}}$ of an inch.

The difference in arrival time was taken to be the difference between the times of occurrence of the peak amplitudes. For the ten tests, the average time it took the waves to travel one foot was $97.76\mu\text{s}$. This means that the approximate speed of the elastic waves inside the pipe is $10,000\text{ft/s}$. This is consistent with the observed speed of shear waves travelling through steel, which is generally around $10,500\text{ft/s}$ [37].

4.1.4 Wave Attenuation Tests

Since acoustic waveforms lose signal strength and information the farther they travel through a medium, tests were carried out to observe the affect that source distance

has on the attenuation of PLB waveforms. First, thee PLB sources were created at 1" increments up to 12" away from the sensor in the longitudinal direction of the pipe.

Figure 4-9 shows the effect that distance had on some of the waveform parameters.

Amplitude (in decibels) and signal strength decreased linearly, while duration, rise time, and counts increased linearly. This is because while the signal gets weaker with distance, it gets spread out and flattened, moving the position of the peak and deforming the threshold based parameters.

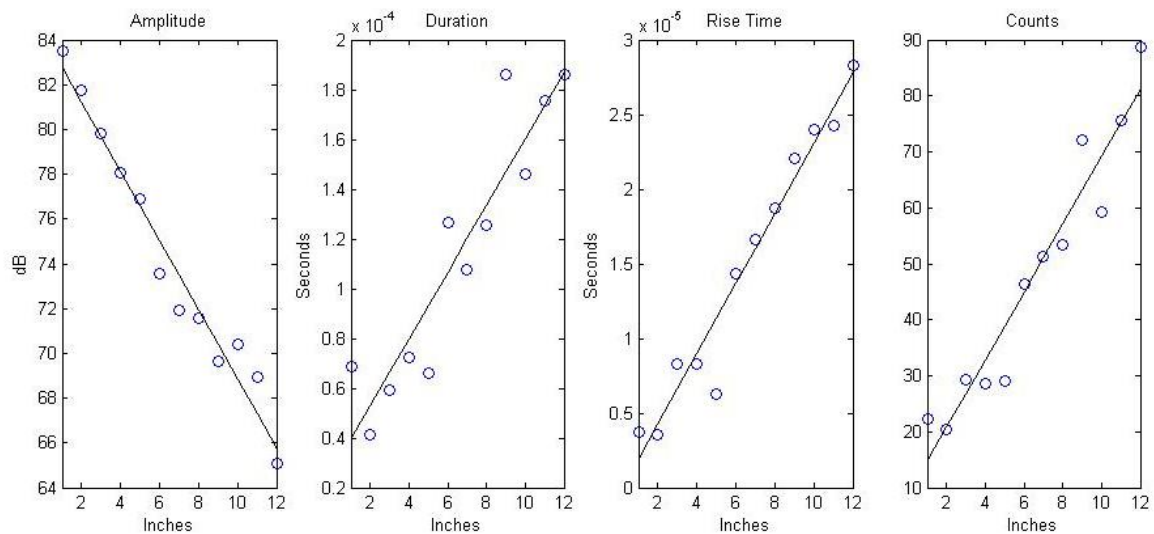


Figure 4-9: Waveform Attenuation Parameters

Figure 4-10 shows the signals and power spectra for PLB sources at 1", 6", and 12" away from the sensor. The signals illustrate the weakening and spreading that occurs over distances due to waveform attenuation. Additionally, the peaks in the power spectra begin to flatten out and lose their information.

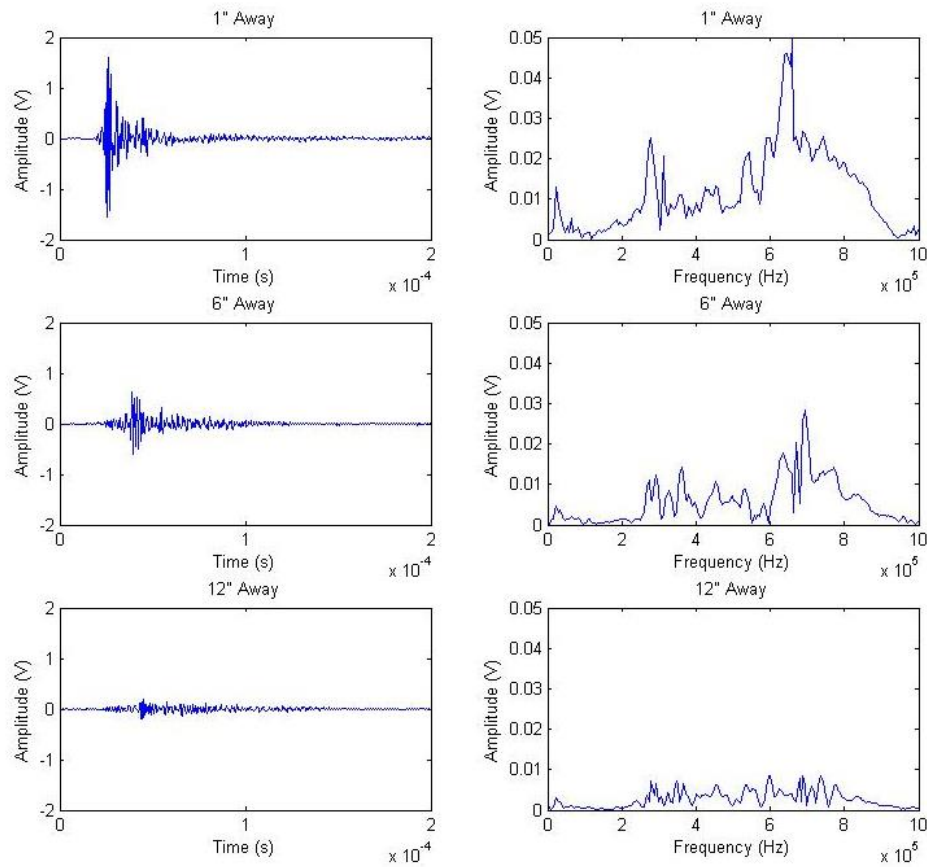


Figure 4-10: Waveform Attenuation Examples, Short

An additional longer range test was carried out, creating 3 PLB sources were averaged at 6" increments for 6', which was almost the entire length of the pipe. The signals show the same continued weakening and spreading. Figure 4-11 shows the signals from waveforms generated 1', 3', and 6' away from the sensor. The same test was carried out at 6" increments in the circumferential direction showing a similar attenuation of waveforms, with the minimum amplitudes occurring 180 degrees away from the sensor.

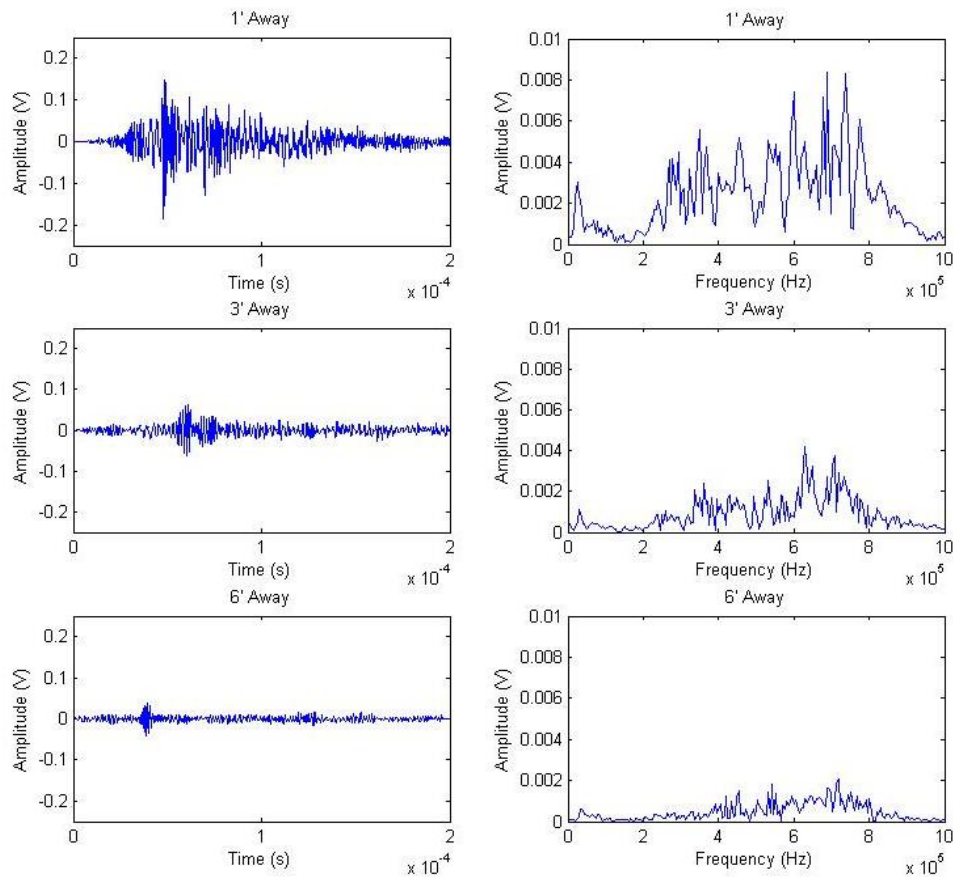


Figure 4-11: Waveform Attenuation Examples, Long

These tests show that there is a significant amount of signal variation due to waveform attenuation when there is distance between the sensor and the transducer. The signal generated from an event 6" away from the sensor has about half the maximum voltage as the signal generated from the same event 1" away from the sensor.

These waveform attenuation tests, as well as the sensor calibration tests performed in section 4.1.1, lead to the creation of the new sensor mount shown in section 3.1.3 which was used in Dent Test 3 detailed in section 4.4.3.

4.2 TENSILE TESTS

In order to test the AE monitoring system, tensile tests were conducted using spare tensile specimen left over from a previous study. These spare tensile specimen reliably produced AE activity at rates in agreement with literature: maximum activity after yield, tapering off to silence before fracture. Unfortunately, the number of cumulative hits varied from sample to sample. Figure 4-12 shows a tensile test with a relatively low amount of acoustic activity, and Figure 4-13 shows a tensile test with a relatively high amount of acoustic activity. The graphs plot the stress-strain curve next to a scatter plot of each hit's amplitude and the strain level at which it was detected.

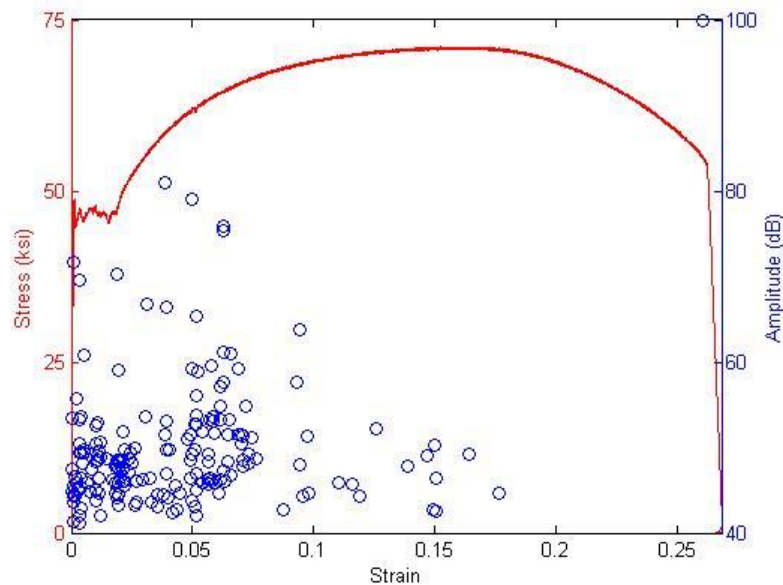


Figure 4-12: Tensile Test – Low Acoustic Activity

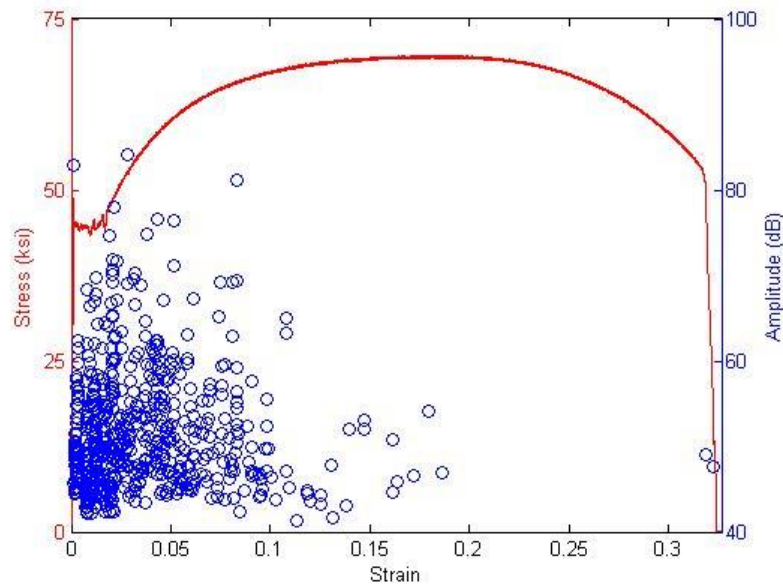


Figure 4-13: Tensile Test – High Acoustic Activity

These tensile tests were conducted on tensile samples manufactured at the same time from the same steel, so it seems unlikely that the differences in activity are mechanical. The threshold levels and analog filters used were the same. The variation could still be due to the fact that sensor-to-surface connectivity between the mounted transducer and the shaved specimen surface varies too much for reliable data acquisition.

Official tensile specimen were manufactured from the same grade of steel as Dent Test 2. Because of scheduling, tensile tests were conducted after the burst and dent tests analyzed in the following sections. It was assumed that AE activity within the tensile specimen could provide insights into activity during the dent tests. However, the API X52 tensile specimen were tested with each analog filter available in the software, and produced an alarmingly small amount of AE activity. Figure 4-14 shows the acoustic activity for one of the API X52 tensile samples.

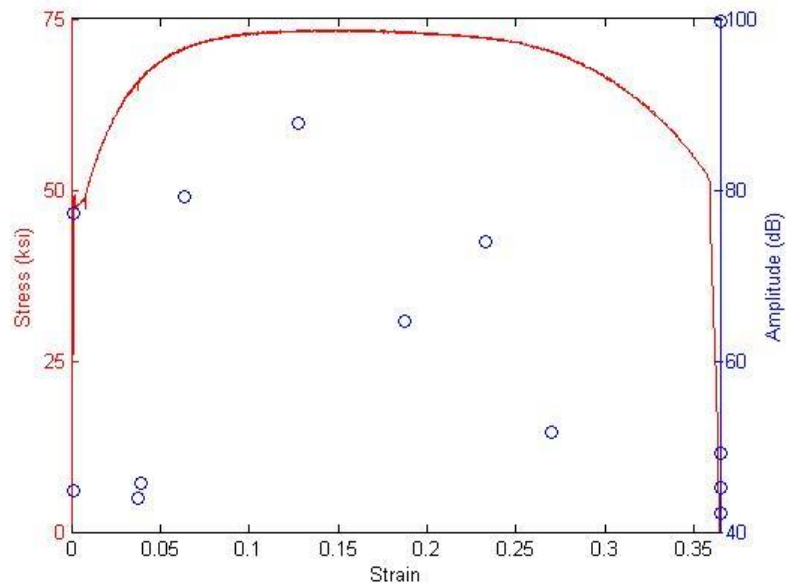


Figure 4-14: Tensile Test – API X52 Activity

Even though the spare tensile samples showed variations in the number of hits detected for each sample, none of them produced anywhere near as few hits as the API X52 samples. It is difficult to say why this is the case, but it may be one of the reasons that Dent Pipe 2—which was also API X52 steel—was not nearly as acoustically active as Dent Pipes 3 and 4 (see section 4.4).

Since critical strain and microvoid nucleation occur after necking and before fracture, and since very little acoustic activity is detected in this range, it may not be possible to use tensile samples to aid in the characterization of acoustic emissions generated at the pipes' critical strain.

4.3 PIPE BURST TEST

As explained in section 3.3, the pipe burst test was carried out very soon after the acquisition of the AE monitoring system and provided a trial run before a pipe dent test was conducted. Since one of the sensors (sensor 2) was removed during the test, the results below focus on the sensor that was attached to the pipe for the entirety of the test (sensor 1). However, for the portion of the test when both sensors were attached, sensor 1 recorded events at almost three times the accumulation rate of sensor 2. Since sensor 1 was located next to the defect that eventually ruptured, this might be an indication of the severity of the defect. Tests carried out in section 4.1.1 suggest that this may instead be an indication that sensor-to-surface contact was inconsistent. Without more burst tests to compare with it is difficult to confirm.

Figure 4-15 shows a scatter plot of each hit's amplitude vs the pipe pressure at its detection time. It gives a good indication of the distribution of activity throughout the burst test. The visible bands correspond to pressure holds because acoustic activity tends to taper off instead of stopping once loading has halted.

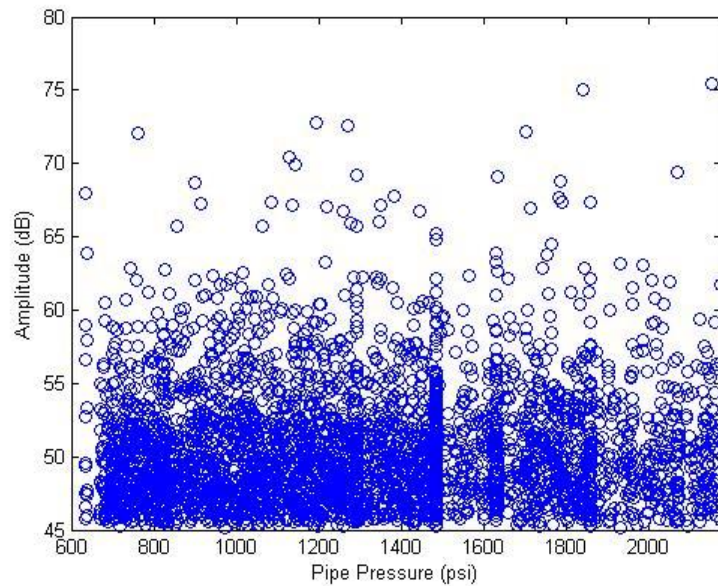


Figure 4-15: Burst Test – Pressure vs. Amplitudes

Figure 4-16 shows the cumulative hits plotted against the pressure of the pipe, and Figure 4-17 shows a zoomed portion of the same plot. Note the similarities between Figure 4-16 and Figure 2-3 in section 2.2.1. When the pressure was relieved and then reapplied about 2/3rds of the way through the test, the sensor barely registered any AE activity which is an example of the Kaiser Effect. Additionally, once the pressure approached the level prior to the hold, the sensor began registering a small amount of AE activity which is an example of the Felicity Effect. Once the pressure exceeded its prior maximum, AE accumulation proceeded as normal.

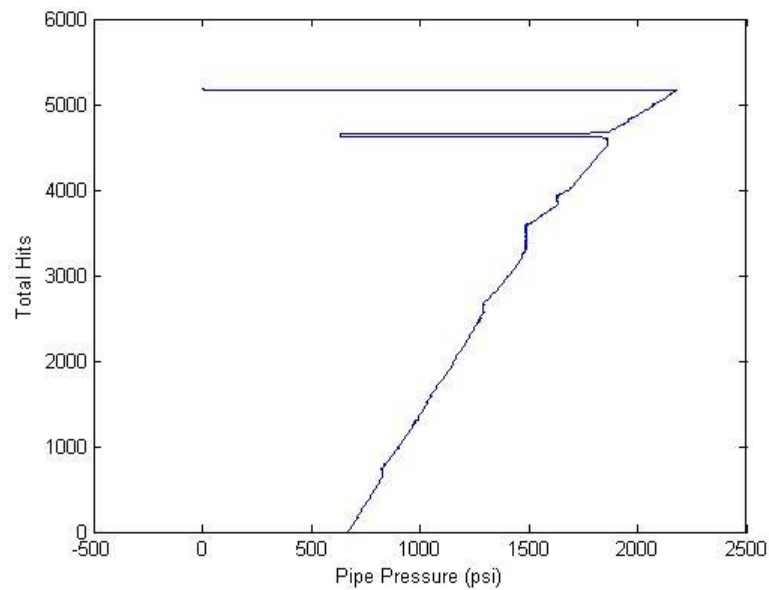


Figure 4-16: Burst Test - Cumulative Hits vs. Pressure

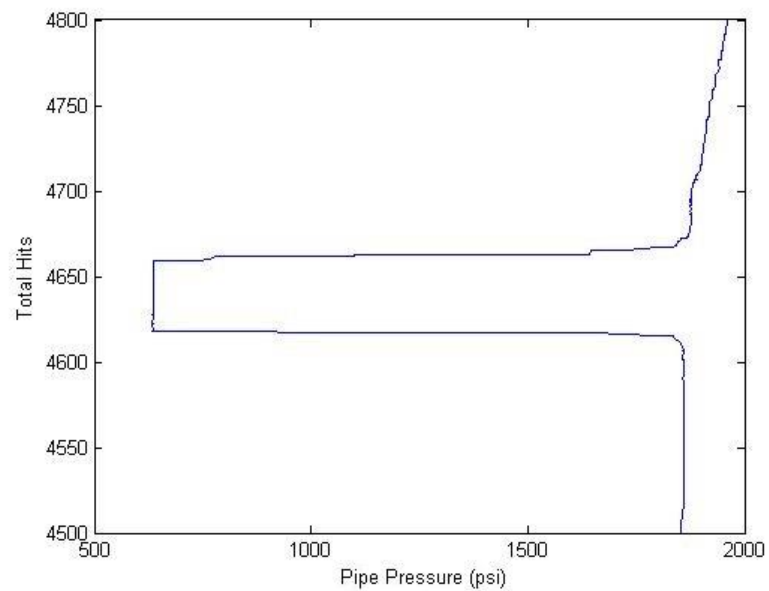


Figure 4-17: Burst Test – Cumulative Hits vs. Pressure (Zoomed)

Figure 4-52 shows the burst test plots of the historic index and the severity, as defined in section 2.4.3. The severity gradually increases as the test progresses because

the total number of relatively high energy events increases with time. While the historic index does deviate too far from unity, it shows that about 2/3rds of the way through the test there was a reduction in the strength of recent activity which picked up again towards the end. This could be an early indication of the impending failure.

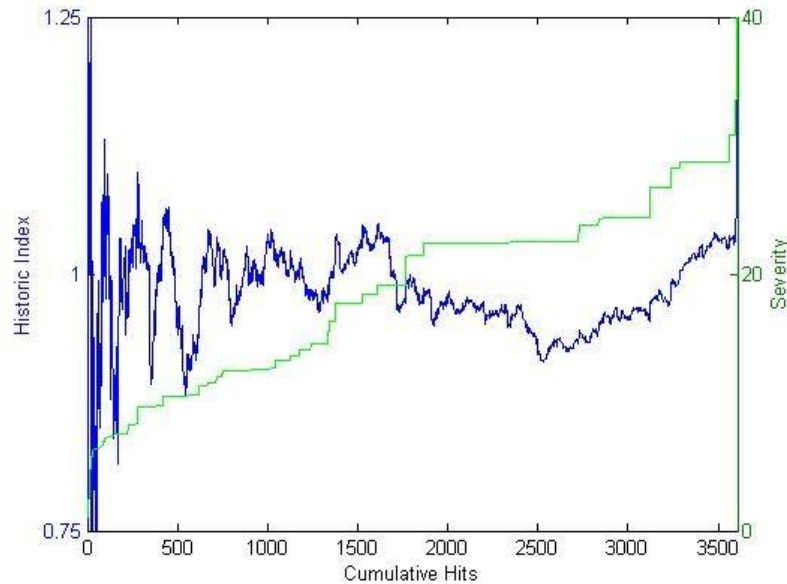


Figure 4-18: Burst Test – Historic Index and Severity

The following pages contain scatter plots of various parameters of individual hits obtained during the burst test. Some of them show data clusters that highlight different types of signals received during the test. With the exception of frequency based values, scatter plot parameters are normalized with 0 being the smallest value observed and 1 being the largest.

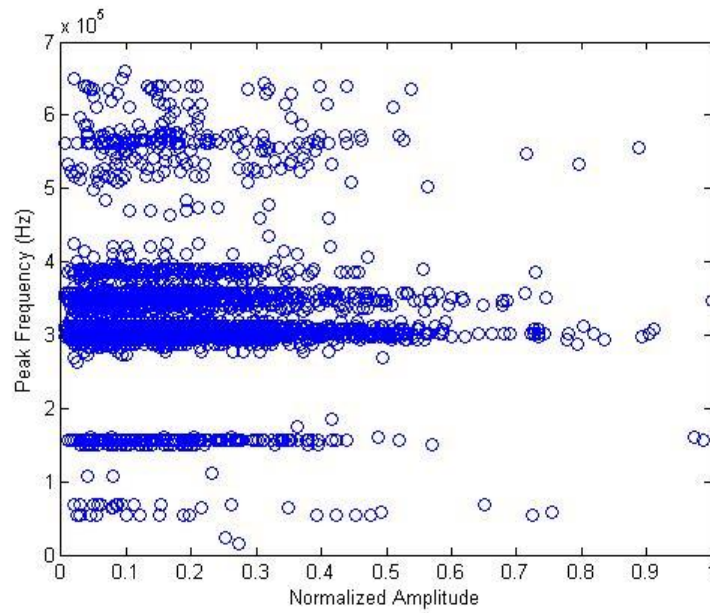


Figure 4-19: Burst Test – Amplitudes vs. Peak Frequencies

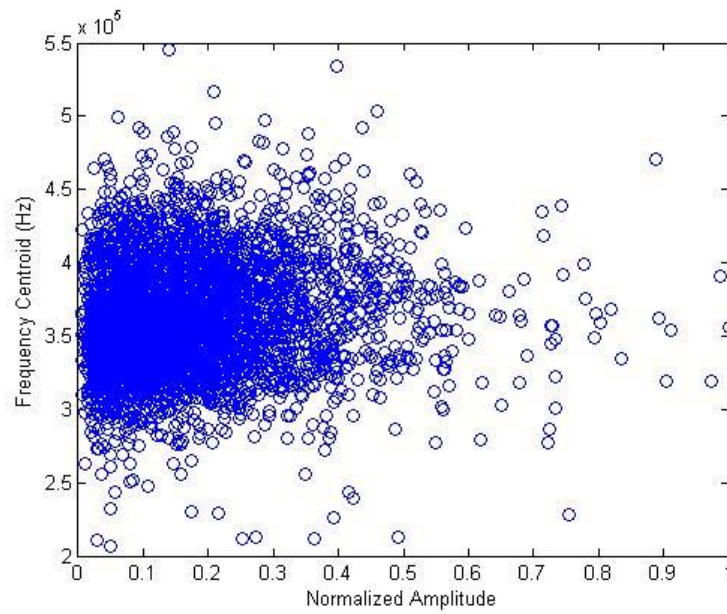


Figure 4-20: Burst Test – Amplitudes vs. Frequency Centroids

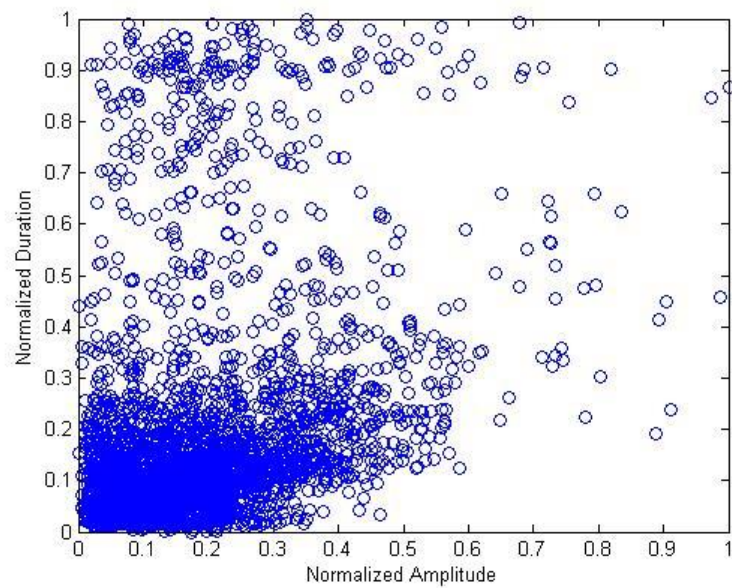


Figure 4-21: Burst Test – Amplitudes vs. Durations

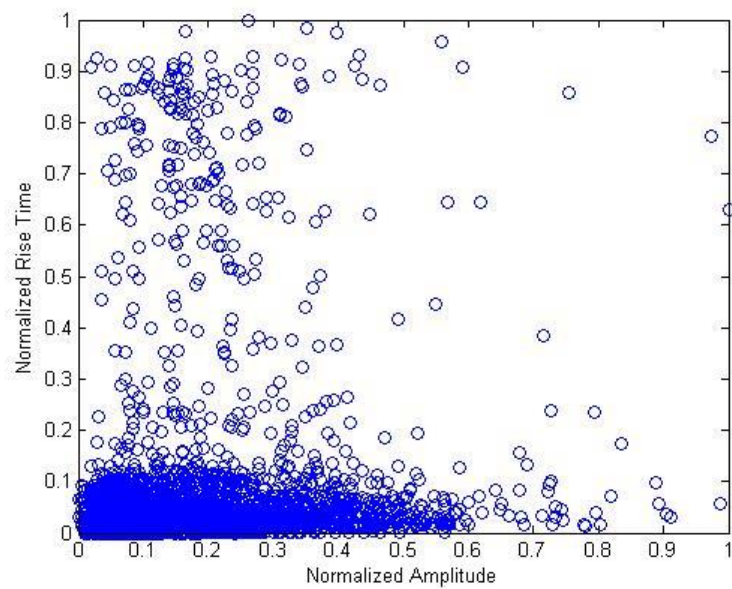


Figure 4-22: Burst Test – Amplitudes vs. Rise Times

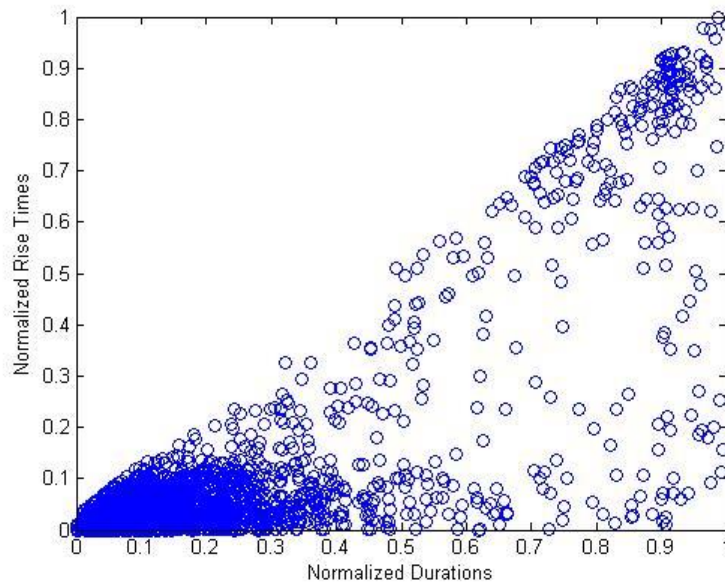


Figure 4-23: Burst Test – Durations vs. Rise Times

Figure 4-19 shows that the event frequency peaks tend to fall within specific bands around 160kHz, 300kHz, 340kHz, 380kHz, and 550kHz. These frequencies are likely those naturally excited in the burst pipe metal. Each individual hit contains multiple frequency peaks which are often at these characteristic values. Some hits have clearly dominant frequencies while others have peaks that are very similar in magnitude. Figure 4-20 shows that when the weighted averages of each hit's power spectrum are plotted, a relatively normal distribution of frequencies arises around 350kHz.

Figure 4-21 and Figure 4-22 show that the majority of hits have short durations and rise times. These are the characteristics of burst type events like the PLB waveforms shown in section 4.1. However, there are clusters of hits with larger timing parameters. While it is self-evident that hits with large rise times must have large durations, a hit with a large duration does not necessarily have a large rise time. But Figure 4-23 reveals

that this is indeed the case for the majority of large duration hits. The triangular population shown in Figure 4-23 is simply due to the fact that a hit's rise time by definition cannot be longer than its duration.

These large duration, large rise time hits have proved to be a standard hit type throughout these pipe tests. They are typically characteristics deformation events occurring relatively far away from the sensors. The analyses in section 4.1.4 show that increasing sensor-to-source distance tends to spread out the waveform and reduce the signal quality. Long duration hits sometimes occur because the ambient sounds in the pipe become loud enough to cause the AE software to register a continuous-type signal. Additionally, sometimes the software picks up two bursts within a single waveform. These can either be due to reflections or simply two separate events that happened near the same time. These dual hits are rare and difficult separate out in post-processing, causing them to be misinterpreted as individual hits with improperly calculated parameters such as duration and rise time. Figure 4-24 shows these different types of long duration hits: from left to right, a continuous signal, a distant source, and a dual hit.

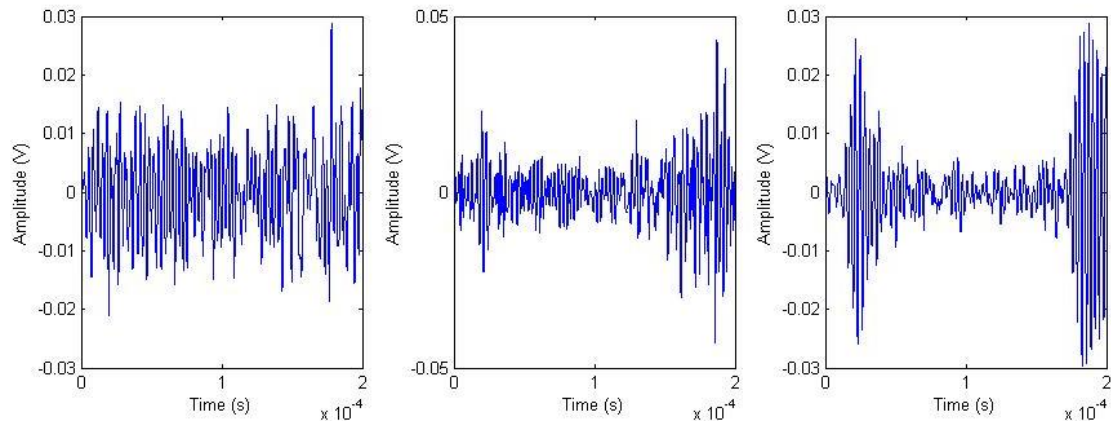


Figure 4-24: Burst Test – Types of Large Duration Hits

These images suggest that the reason the large duration, large rise time hits tend to cluster together as shown in Figure 4-23 is simply because the software only records waveforms up to a specific length. It is clear that the true durations of these signals are often larger than those permitted in analysis with this software. If the true durations were plotted against their rise times, it is likely that the cluster towards the top of Figure 4-23 would spread out.

4.4 PIPE DENT TESTS

As described in section 3.4, the three pipe dent tests conducted during this research were different for a number of reasons. Each dent was created at a different location on the pipe and Dent Pipe 2 was a different size and steel grade than the other two. The dents in Dent Tests 2 and 3 were placed on welds, while the dent in Dent Test 1 was on the pipe body. Dent Test 3 was approached with a far greater knowledge of the relationships between event waveform and sensor signal quality because it was scheduled much later than Dent Tests 1 and 2. All three dent tests are analyzed in the following sections, but it is clear upon comparison that Dent Test 3 was the most successful in terms of the measurement of meaningful AE activity from the dent and subsequent crack.

4.4.1 Dent Test 1

Dent Test 1 reached the deepest depth of the three tests, and it was the only test that did not result in a crack. The test was halted once it was apparent that the dent profile was not changing much and the deformation was occurring in other areas such as the sides of the pipe and the area around the saddle. As explained in section 3.4.1, sensor 1 was placed on top of the pipe and sensor 2 was placed on the bottom of the pipe. They were both set with the 100-600kHz filter and a threshold of 40db.

During the course of the test, sensor 1 registered 3917 hits and sensor 2 registered 3559 hits. After the post-processing filters described in section 3.1.2 were applied, they were left with 1912 and 2073 respectively. Figure 4-25 shows the rates of accumulation of hits vs. the dent depth.

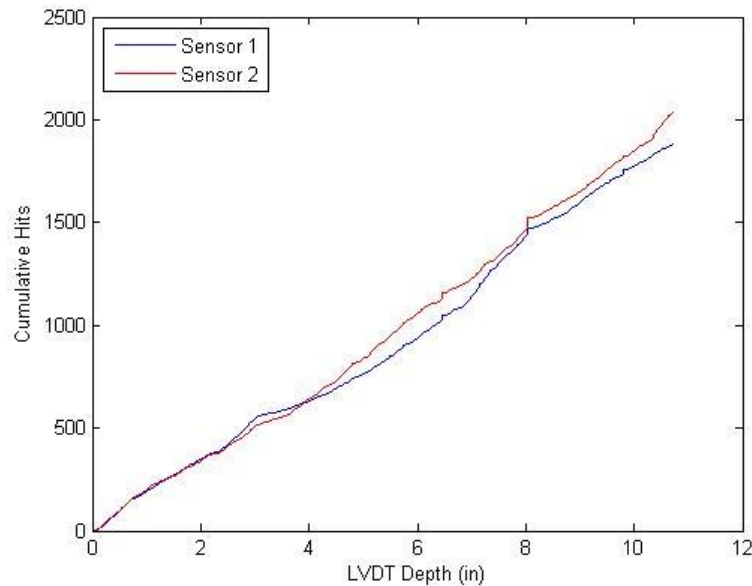


Figure 4-25: Dent Test 1 – Cumulative Hits

Figure 4-26 shows a scatter plot of the event amplitudes and the depths at which they were detected. While Figure 4-25 shows fairly linear rates of accumulation, Figure 4-26 shows that the distributions of intensities are more spread out throughout the test. Between depths of 5" and 8" the sensor on top was registering a large amount of high amplitude events. But after 8", high amplitude activity began to drop off at the top sensor and pick up at the bottom sensor. This may be an indication that once the dent depth was deep enough, the pipe began to deform more near its support than near the dent.

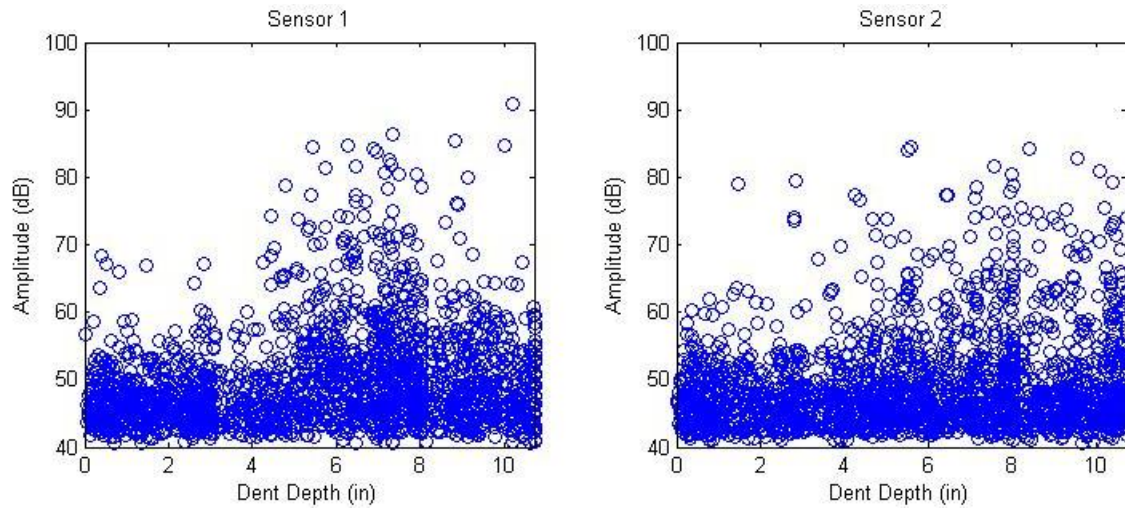


Figure 4-26: Dent Test 1 – Depths vs Amplitudes

Figure 4-27 shows the load cell and the strain gauges plotted against the LVDT depth. Refer to section 4.3 for information on the placement and names of these strain gauges. The strains for SGH1 were significantly larger than the strains for the other three gauges, so in order to more effectively show their behavior on the same graph, the strains for SGH1 have been reduced to 1/3rd their true value.

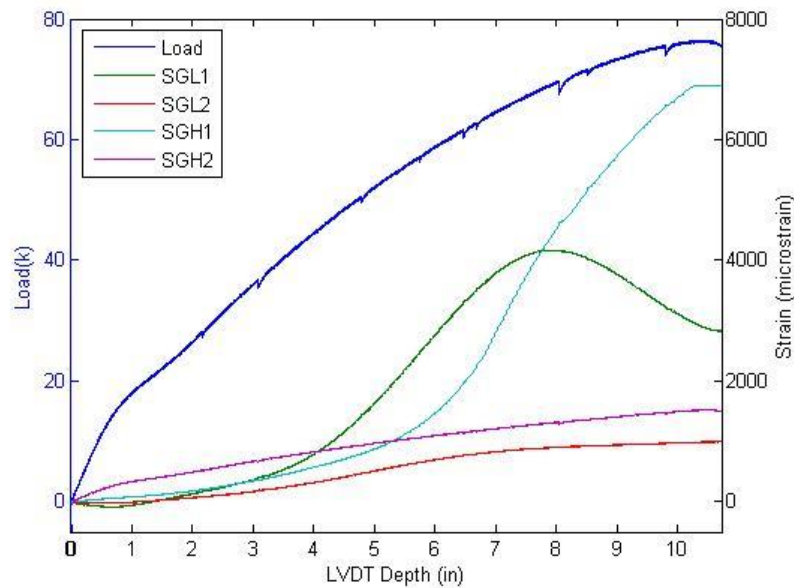


Figure 4-27: Dent Test 1 – Instrumentation

Note the similarities between the amplitude distribution shown for sensor 1 in Figure 4-26 and the strain curve shown by SGL1 in Figure 4-27. SGL1 was 13" away from the dent apex in one direction and sensor 1 was 12" away in the opposite direction. Since the deformation of the pipe is symmetrical, SGL1 gives a good indication of the strains directly underneath sensor 1. Between 5" and 8" depth the strains in SGL1 were rapidly increasing and sensor 1 was measuring a lot of high amplitude events. After 8", the strains in SGL1 actually started decreasing and sensor 1 was not picking up as many high amplitude events.

Even though the majority of high amplitude activity recorded by sensor 1 stopped after 8", the strongest signal was acquired at a depth of 10". Figure 4-28 shows the signal of this hit.

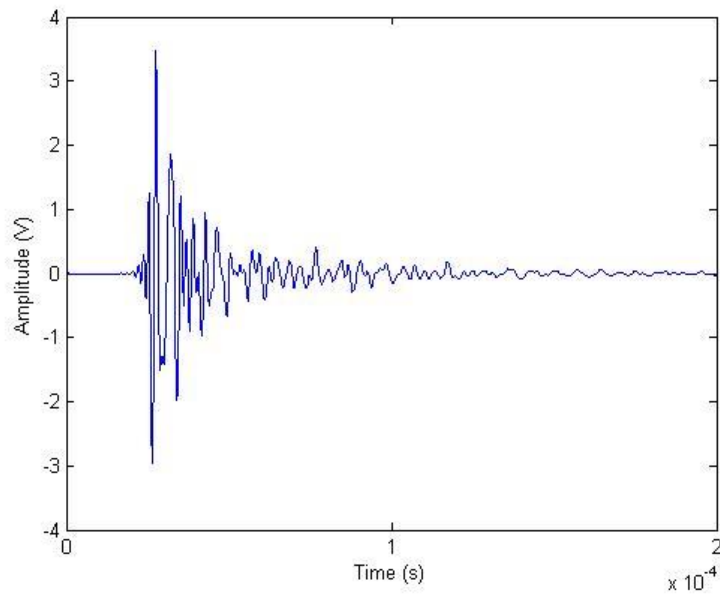


Figure 4-28: Dent Test 1 – Strongest Signal

Figure 4-29 shows the cumulative signal strength curves for each sensor. Figure 4-30 shows the historic index and severity curves for each sensor, which help further explain the trends in signal strength. The CSS curve for sensor 1 seems to match the cumulative hits curve in Figure 4-25. But the historic index shows a bulge corresponding to the period of high activity between 5" and 8".

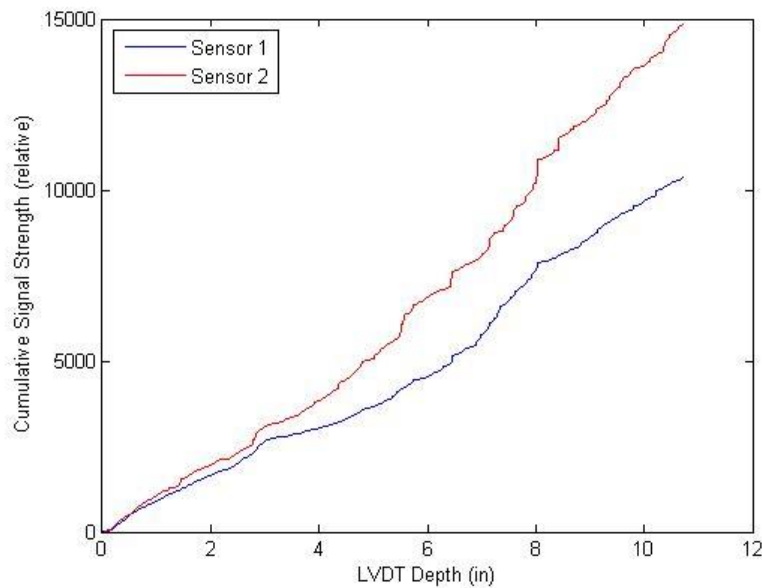


Figure 4-29: Dent Test 1 – Cumulative Signal Strength

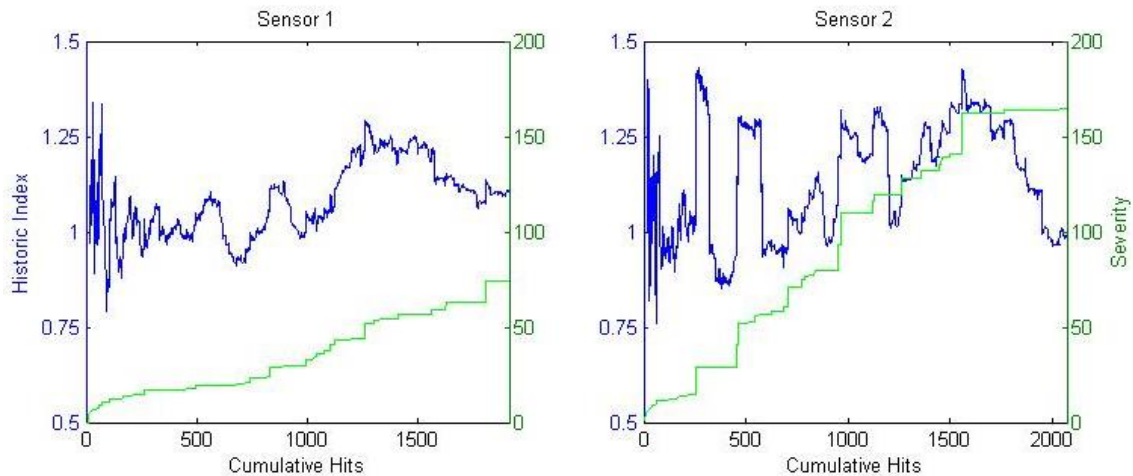


Figure 4-30: Dent Test 1 – Historic Index and Severity

While CSS curve for sensor 2 gradually increased over the course of the test as more high amplitude events were registered, its historic index shows there were actually some sharp spikes in local activity earlier in the test. These are shown in Figure 4-26 as the handful of sensor 2 signals with amplitudes greater than 70db that were registered at

depth of 5" at sensor 2. Sensor 1 did not register any signals of that magnitude before 5", and the historic index is shown to be less wildly varied earlier in the test.

The following pages contain scatter plots of various parameters of individual hits obtained during the burst test. Some of them show data clusters that highlight different types of signals received during the test. With the exception of frequency based values, scatter plot parameters are normalized with 0 being the smallest value observed and 1 being the largest.

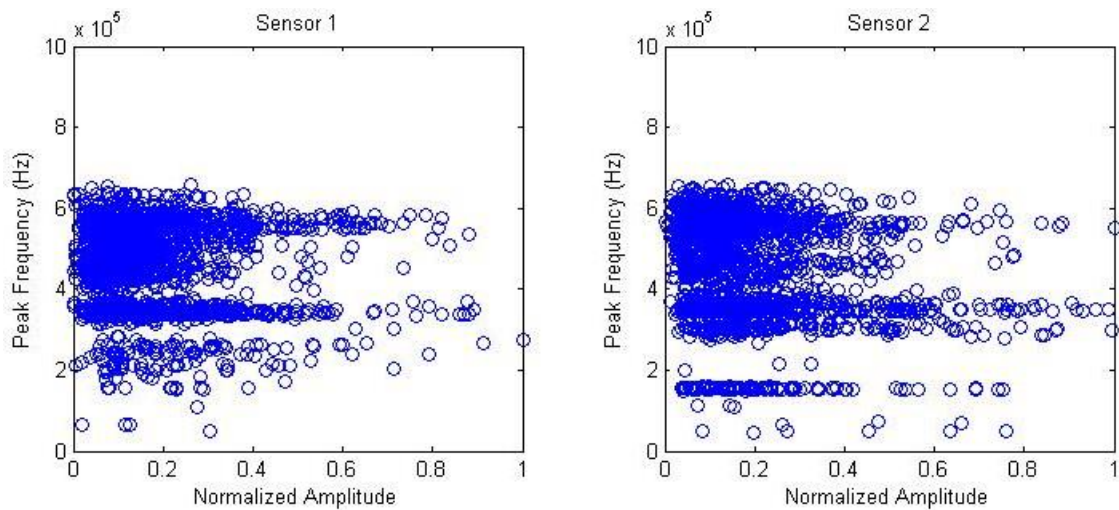


Figure 4-31: Dent Test 1 – Amplitudes vs. Peak Frequencies

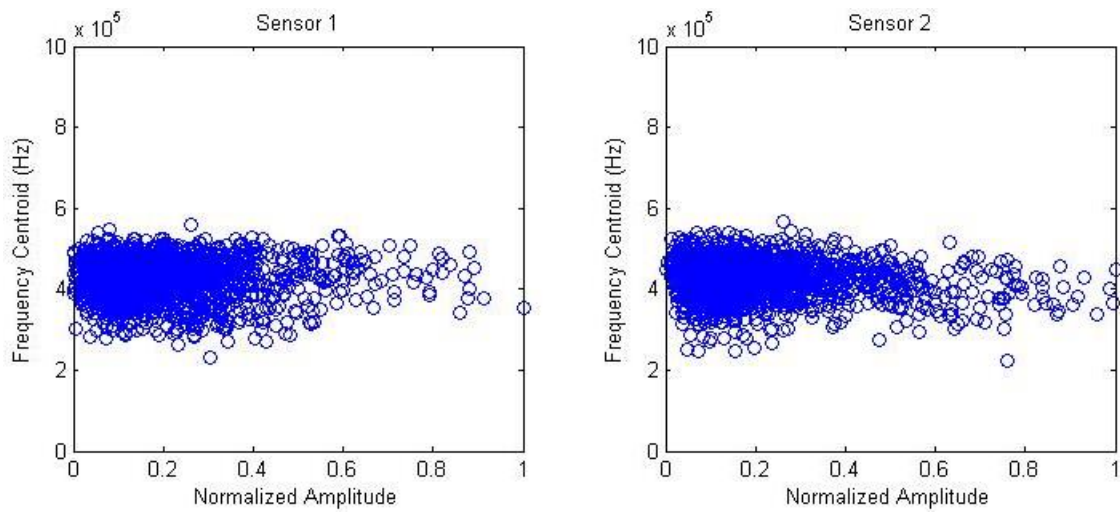


Figure 4-32: Dent Test 1 – Amplitudes vs. Frequency Centroids

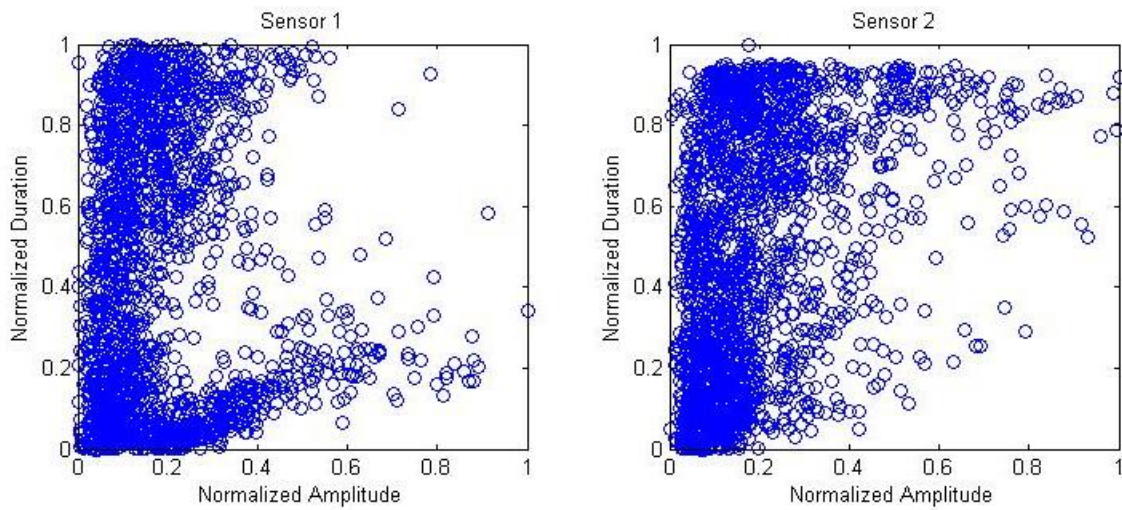


Figure 4-33: Dent Test 1 – Amplitudes vs. Durations

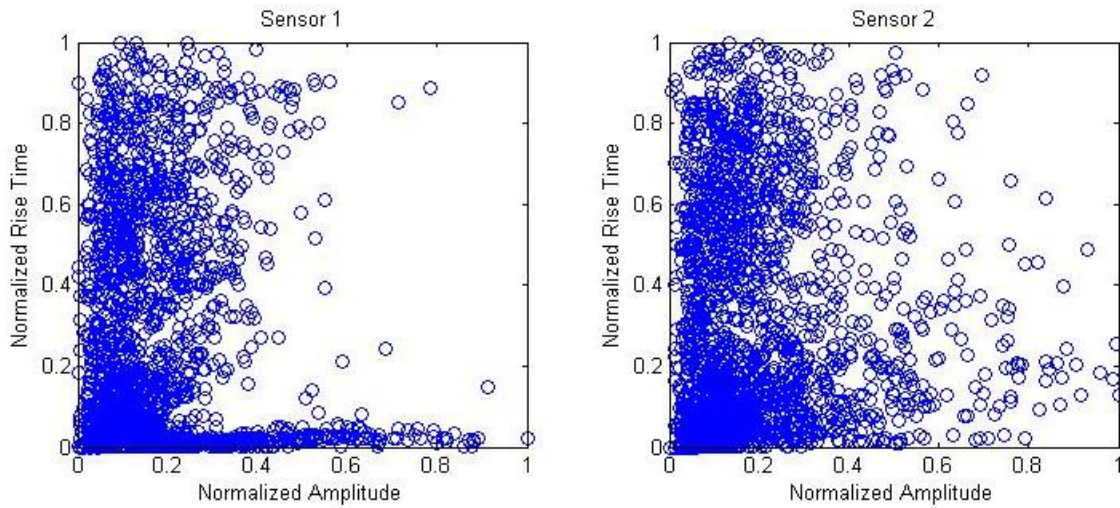


Figure 4-34: Dent Test 1 – Amplitudes vs. Rise Times

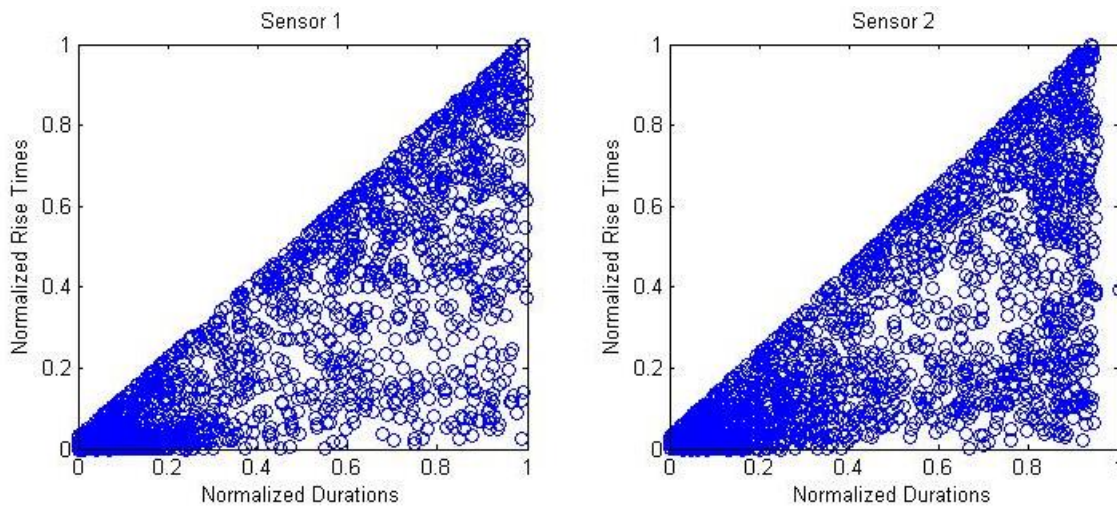


Figure 4-35: Dent Test 1 – Durations vs. Rise Times.

Just like in the burst test, Figure 4-31 shows that the peak frequencies tend to fall somewhat randomly within specific bands, but Figure 4-32 shows that the averaged frequency content of all signals reveals a normal distribution of excitation. Sensor 2 registered a band of signals with peak frequencies around 180kHz, which did not appear at sensor 1. After grouping these signals and investigating their waveforms, none of

them are sharp burst events, they are all longer duration messy hits. Perhaps the bottom sensor was picking up a noise frequency from the support that ended up being the dominant frequency in some of the more noisy hits.

Figure 4-33 shows that sensor 1 picked up a cluster of signals with large amplitudes and short durations which sensor 2 did not. Additionally, Figure 4-34 shows another cluster of high amplitude, short rise time signals for sensor 1 but not for sensor 2. Even though Figure 4-35 shows a fairly random and even grouping of rise times and durations, it is clear by overlapping the sensor 1 plots from Figure 4-33 and Figure 4-34 that the high amplitude events with short rise times are also the ones with short durations.

Hits with short rise times and short durations are by definition sharp burst type events. Based on previous analyses, the closer the signal is to the event source, the sharper the signal burst generated is. So these sharp bursts signal are an indication of relatively intense deformation activity occurring close to sensor 1.

For each sensor, hits with amplitudes greater than 50db durations shorter than 50 μ s were grouped together for observation. Figure 4-36 shows the distribution of amplitudes and their depth of detection for this subgroup of hits. It is clear that deformation near sensor 1 (12" away from the dent apex) was at its peak between 5" and 8", while deformation near the sensor 2 (next to the support saddle underneath the pipe) was beginning to increase when the test was stopped.

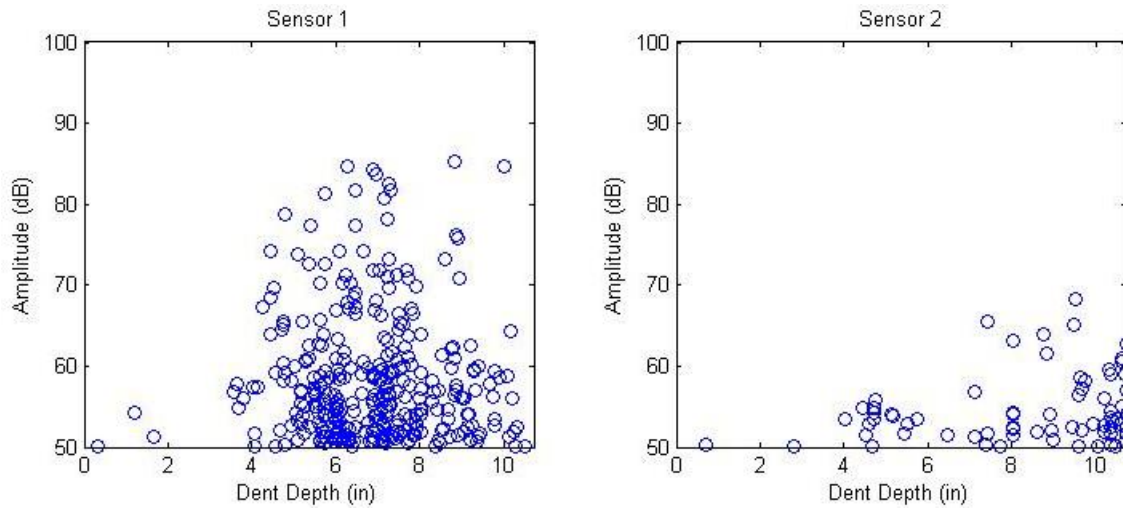


Figure 4-36: Dent Test 1 – Depths vs. Amplitudes for Sharp Hits

The behavior shown in the sensor 1 plot is further corroborated by the SGL1 curve in Figure 4-27. While there was no strain gauge to reference sensor 2 with, SGH1 reveals that when longitudinal deformation around the dent was slowing down, circumferential deformation was drastically increasing. Based on the analyses above, it is likely that this corresponded to an increase in acoustic activity occurring near SGH1. This would have manifested itself as an increase in high amplitude, sharp events at deeper depths had there been a transducer placed near SGH1. Inversely, had there been a strain gauge underneath the pipe, the slope of its strain curve would have likely been increasing towards the end of the test as indicated by the incremental number of strong, sharp signals registered by sensor 2.

4.4.2 Dent Test 2

A surface crack was visible on Dent Test 2 after only 2" of depth. By 4", the crack ripped open and the test was stopped. The sensors were in the same positions and using the same settings as Dent Test 1 since the AE monitoring system performed well. Unfortunately, despite all the deformation activity in Dent Test 2, sensor 1 only registered 333 hits and sensor 2 registered 1017 hits. After the post-processing filters described in section 3.1.2 were applied, they were left with 157 and 494 respectively. Figure 4-37 shows the cumulative hits for each sensor plotted against the LVDT depth, and Figure 4-38 shows the scatter plot of event amplitudes and their depths of detection. Figure 4-39 shows the load cell and the strain gauges plotted against the LVDT depth. Refer to section 4.3 for information on the placement of these strain gauges.

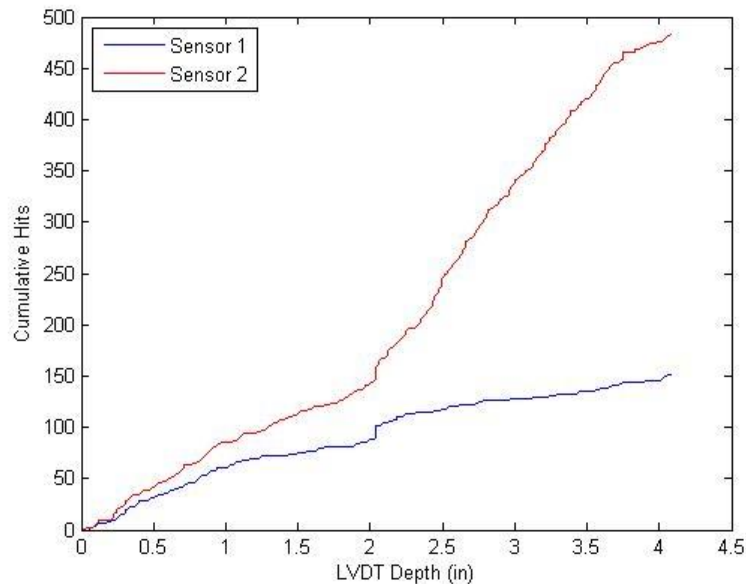


Figure 4-37: Dent Test 2 – Cumulative Hits

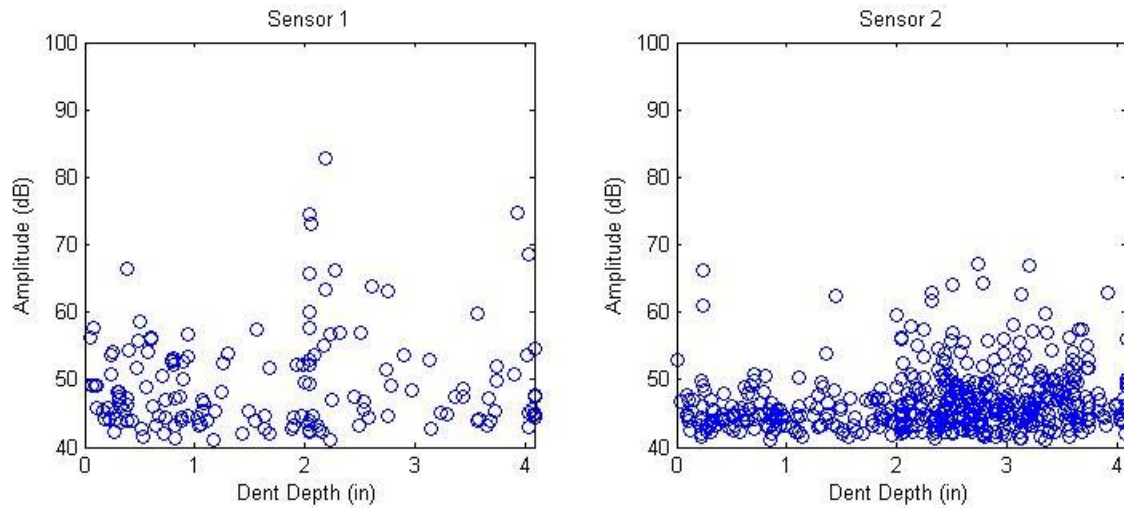


Figure 4-38: Dent Test 2 – Depths vs Amplitudes

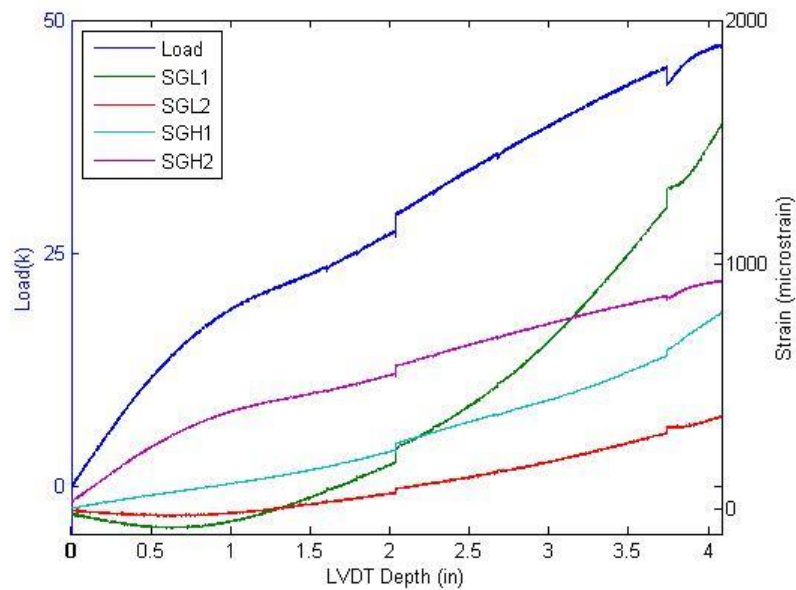


Figure 4-39: Dent Test 2 – Instrumentation

The spikes shown in Figure 4-37 and Figure 4-39 occurred when the test was paused for lunch on the first day of testing (at 2" depth) and then again when the test had to be paused overnight (around 3.75" depth). The pipe must have relaxed or adjusted somewhat underneath the static actuator, causing variations in the loads and strains.

When sensor 1 was moved closer to the dent apex while the crack was growing, it did begin registering some higher energy events. These are seen in Figure 4-38 after a depth of 2". Sensor 2 also saw a spike in activity around this time. But even though the crack was visibly growing on the video monitor, sensor 1's accumulation slowed to a trickle, and it barely registered any events when the crack ruptured.

While they do not contain as much useful information as the other tests, the following pages contain figures displaying information on the cumulative signal strength and scatter plots of hit characteristics for Dent Test 2. With the exception of frequency based values, scatter plot parameters are normalized with 0 being the smallest value observed and 1 being the largest.

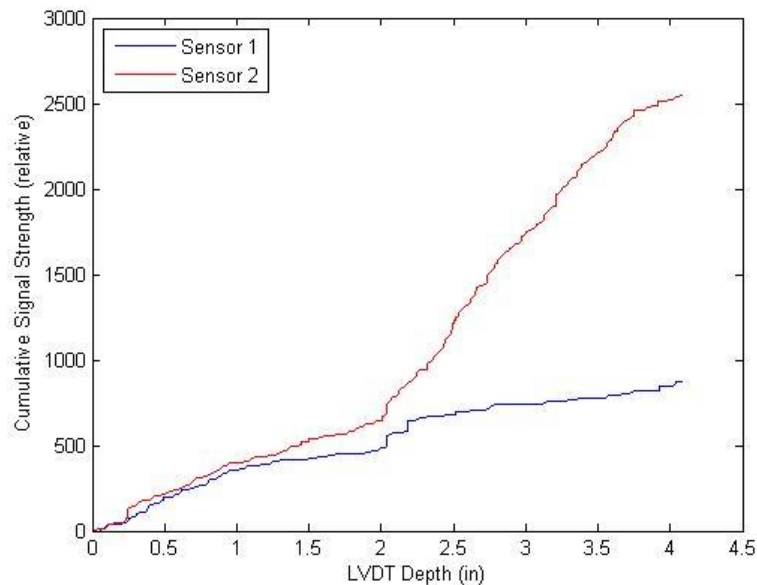


Figure 4-40: Dent Test 2 – Cumulative Signal Strength

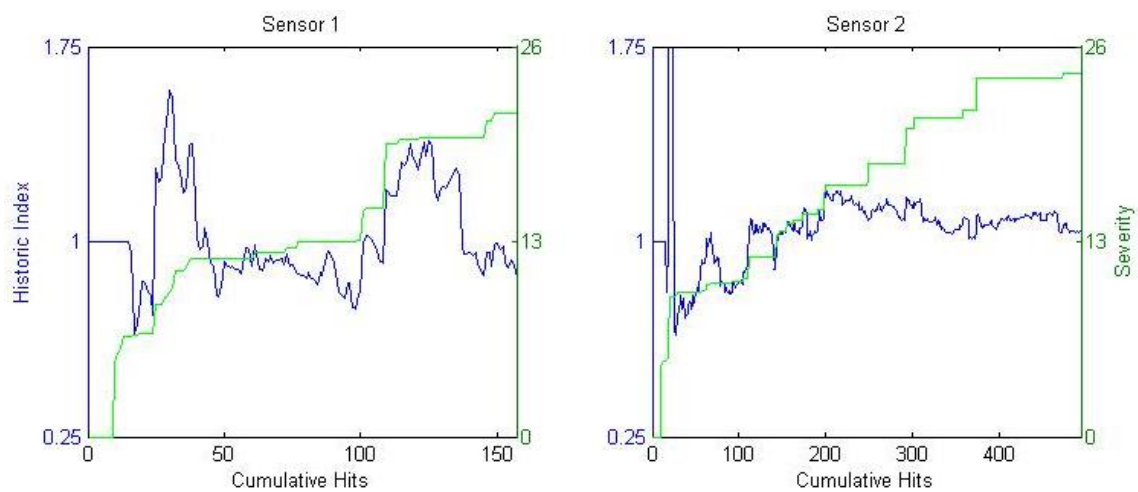


Figure 4-41: Dent Test 2 – Historic Index and Severity

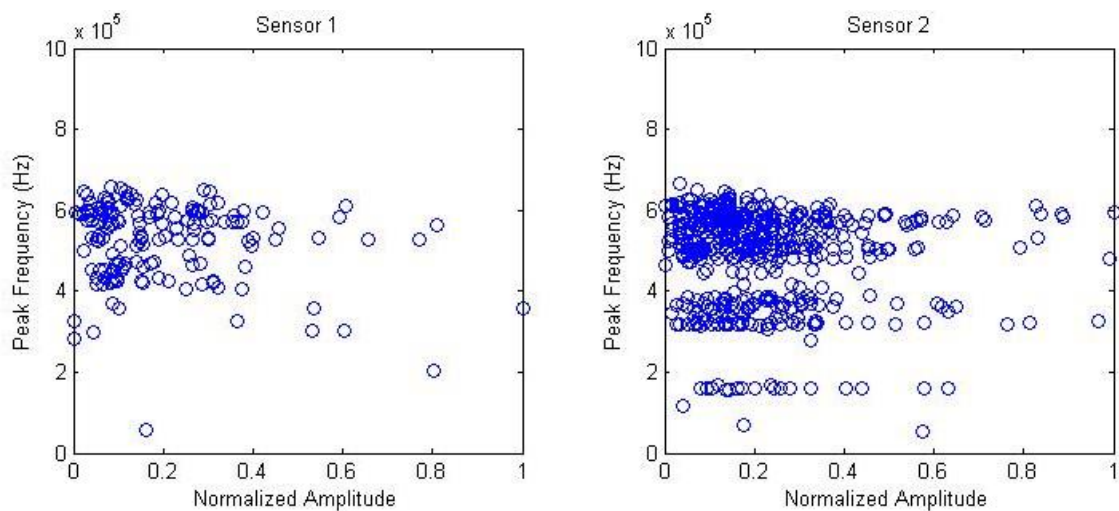


Figure 4-42: Dent Test 2 – Amplitudes vs. Peak Frequencies

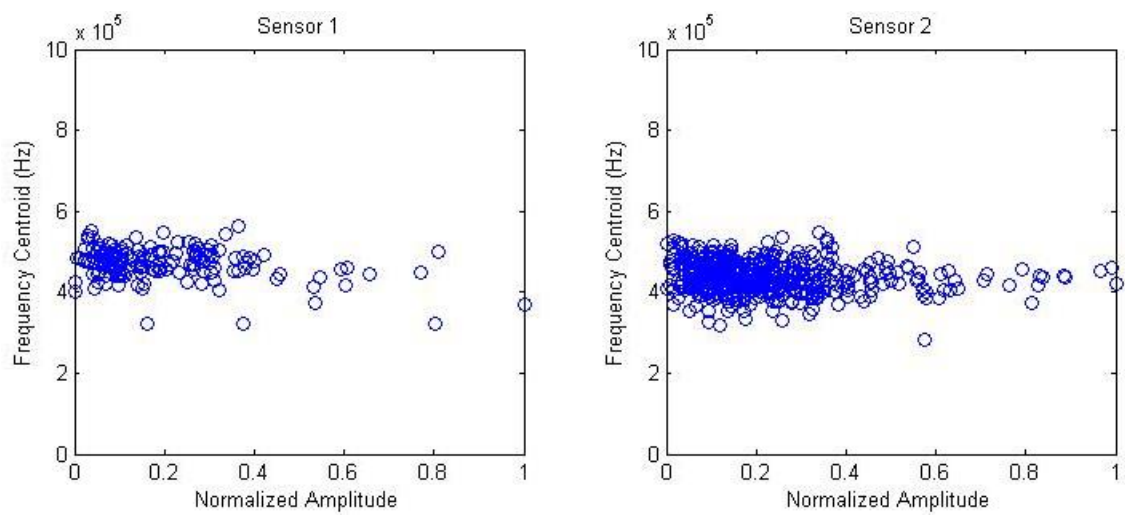


Figure 4-43: Dent Test 2 – Amplitudes vs. Frequency Centroids

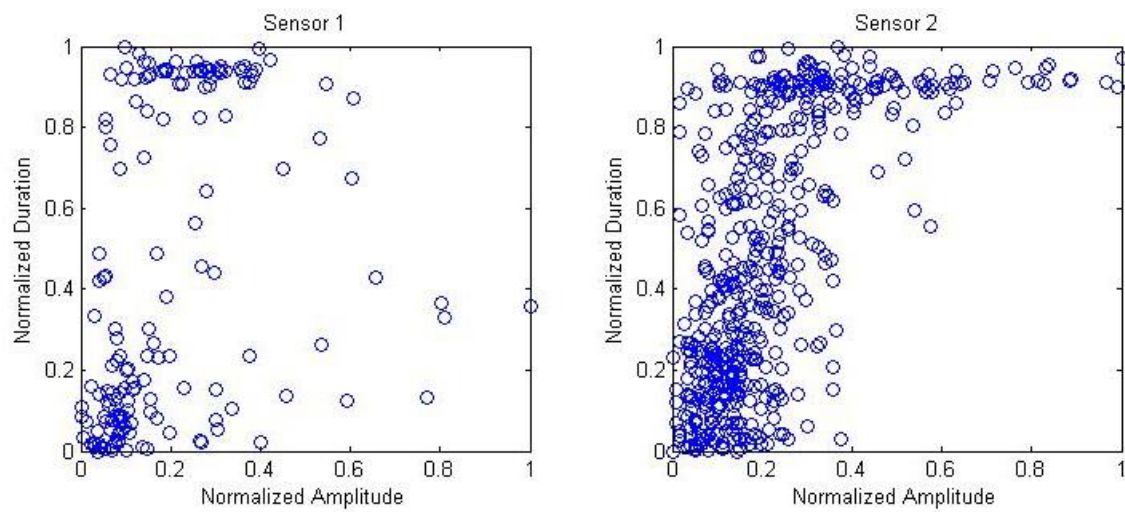


Figure 4-44: Dent Test 2 – Amplitudes vs. Durations

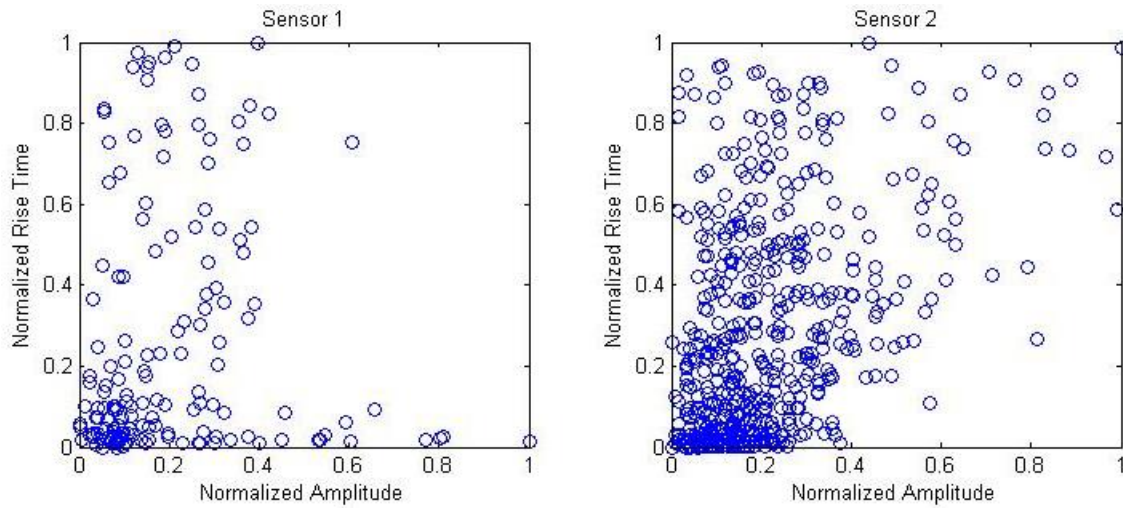


Figure 4-45: Dent Test 2 – Amplitudes vs. Rise Times

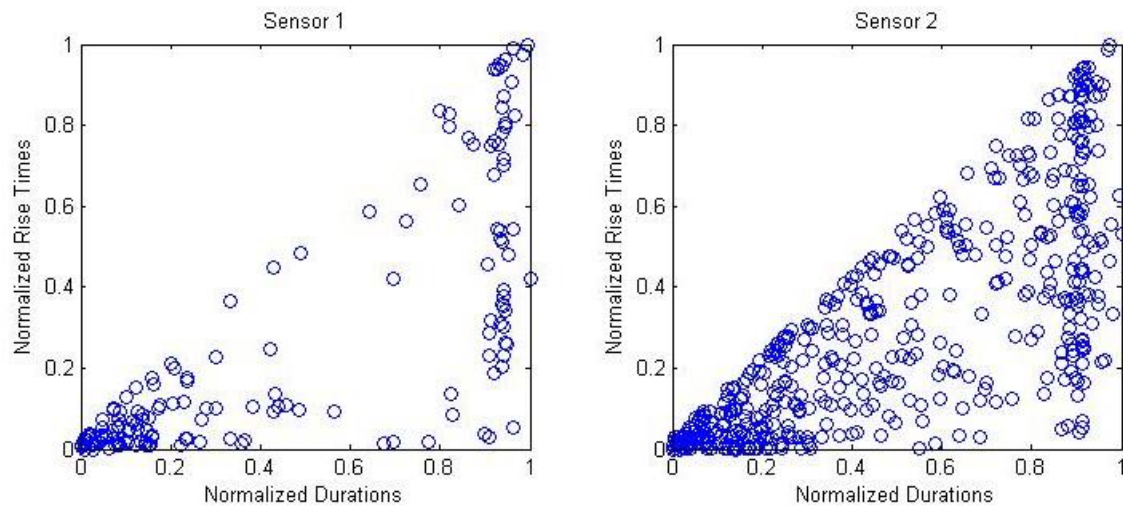


Figure 4-46: Dent Test 2 – Durations vs. Rise Times

Between Dent Tests 2 and 3, efforts described in section 4.1 were made to determine the cause of the failure of the AE monitoring system during Dent Test 2. Tests identified that the sensor-to-surface contact quality may have been an issue, especially once sensor 1 was placed close to dent when the curvature was high.

It was also found that a sensor-to-source distance of 12" causes significant waveform attenuation and a weakened and distorted signal. This may also be the reason that signal activity at sensor 1 in Dent Test 1 was most intense when the strains at SGL1 were rapidly increasing, not when the strains at the dent apex were. Even though the AE monitoring system recorded a large amount of interesting data in Dent Test 1, it could have undervalued some significant waveforms created closer to the dent apex. The new transducer mount shown in section 3.1.3 were made and used for Dent Test 3. These mounts maintained better sensor-to-surface contact than the magnetic clamps and allowed the sensors to be placed closer to the dent apex.

Even though these improvements to the sensor mounts clearly improved the quality of recorded activity, the analysis in section 4.2 shows that the relatively quiet behavior of Dent Pipe 2 could simply be due to the fact that it was made out of API X52 steel while Dent Pipes 1 and 3 were made out of API X65. For some reason the API X52 tensile specimen produced virtually no acoustic activity measurable by the monitoring system. It is difficult to say exactly why this is the case, but it may have something to do with the frequency content of the API X52 waveforms and the limitations of the analog filters used in this study.

While there are no studies that note this lack of acoustic activity in API X52, Budano et al. measured AE in API X65, API X80, and API X100 steel tensile specimen and noted large variations between the cumulative hit counts for each type of metal, with API X65 being the noisiest [27]. The chemical analysis report in section 3.4 shows differences in the carbon and silicon contents of the API X52 and API X65 steels which

is likely connected to the acoustic discrepancy between the two steels. Further research may have to be conducted to discover why the acoustic behavior of the API X52 was different than the other steels as shown in this study.

4.4.3 Dent Test 3

Unlike Dent Tests 1 and 2, both sensors were placed on top of Dent Pipe 3 and set with different analog filters. Sensor 1 was set with the 100-600kHz filter and a threshold of 50db. Sensor 2 was set with a 400-990kHz filter and a threshold of 35db. As explained in section 3.4.3 this was done to make all attempts to capture acoustic activity during crack growth.

During Dent Test 3 sensor 1 registered 938 hits and sensor 2 captured 21407 hits. This disparity is primarily due to their different thresholds which were 50db (33mV) and 35db (6mV) respectively. For the analysis of Dent Pipe 3, all hits below 50db were removed during post-processing. This was because the low energy hits picked up by sensor 2 were registered at a constant rate throughout the test and do not provide much information about the deformation. It also allowed for a more direct comparison between the two sensors. Even though the software threshold were different for Dent Test 3, threshold based parameters were still calculated using the post-processing floating threshold described in section 3.1.2.

After the post-processing filters described in section 3.1.2 were applied, sensor 1 had 805 hits and sensor 2 had 1582 hits. The reduction in hits registered by sensor 1 was

due to the fact that the AEwin™ software still somehow registered a few hits below the threshold of 50db. Figure 4-47 shows the cumulative AE hits for each sensor plotted against the depth.

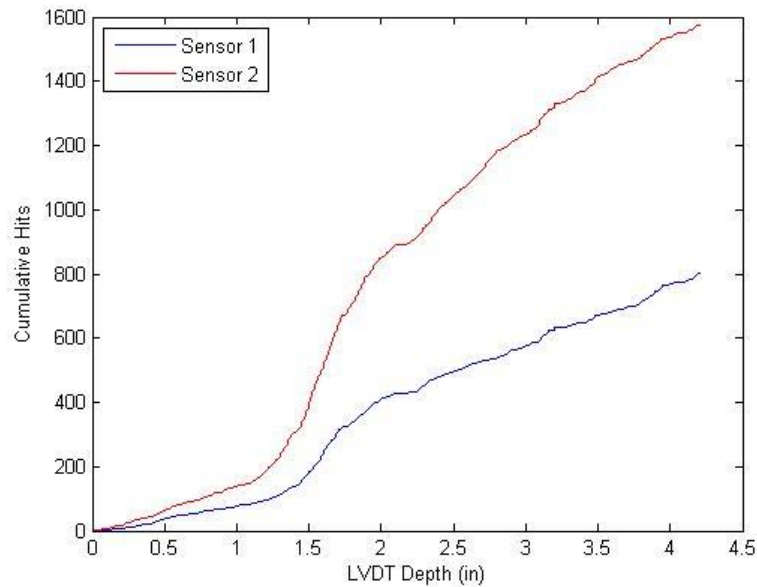


Figure 4-47: Dent Test 3 – Cumulative Hits

Figure 4-48 shows a scatter plot of each hits amplitude and its corresponding depth of detection. Note that the rapid increase in cumulative hits between 1" and 2" shown in Figure 4-47 corresponds to the cluster of higher energy events shown in Figure 4-48. This is also around the depth at which the crack next to the girth weld was first noticed, so this trend may be linked to crack formation.

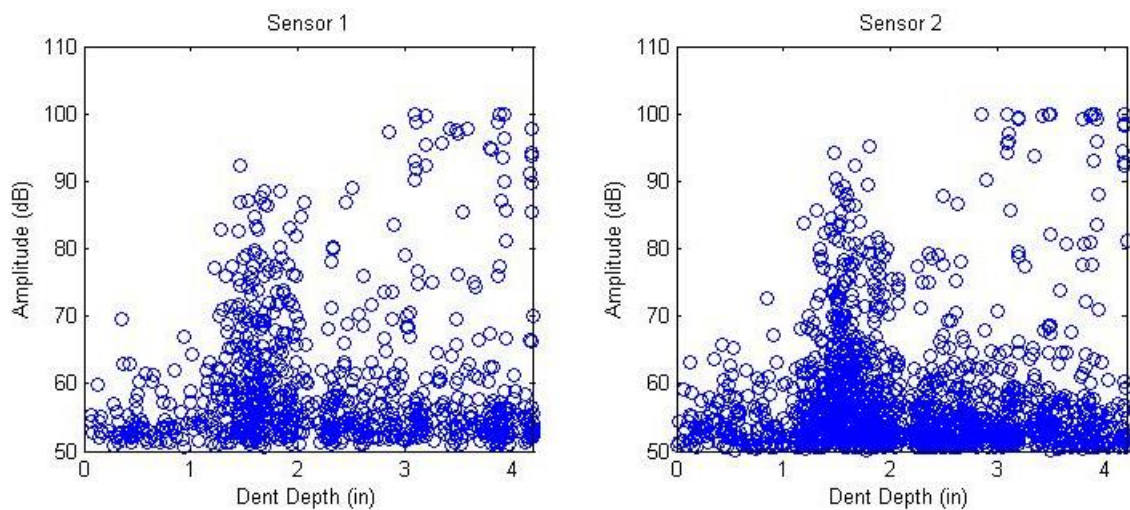


Figure 4-48: Dent Test 3 – Depths vs. Amplitudes

After that cluster of strong events, the pipe remained relatively quiet for period of time. Then around a depth of 3" the sensors began registering very high energy events coming from the dent apex. At this point the crack was already developed and was visibly seen gradually expanding on the video monitor. Most of these high energy events were quite audible as large metallic *pings* coming from the pipe. Some of them also resulted in visible vibrations on the video monitor. At least one of was strong enough to cause some contrast paint chips to fall off the edge of the crack. It is clear based on observation that these high energy acoustic events correspond to crack growth.

Figure 4-49 shows the load cell and the strain gauges plotted against the LVDT depth. Refer to section 4.3 for information on the placement of these strain gauges. Around a depth of 1", the load starts dropping off as parts of the pipe begin to yield. This corresponds to an increased strain rate in SGL1 and SGH1 which were closer to the dent

apex. This is also close to the depth when high energy events began rapidly occurring as shown in Figure 4-48.

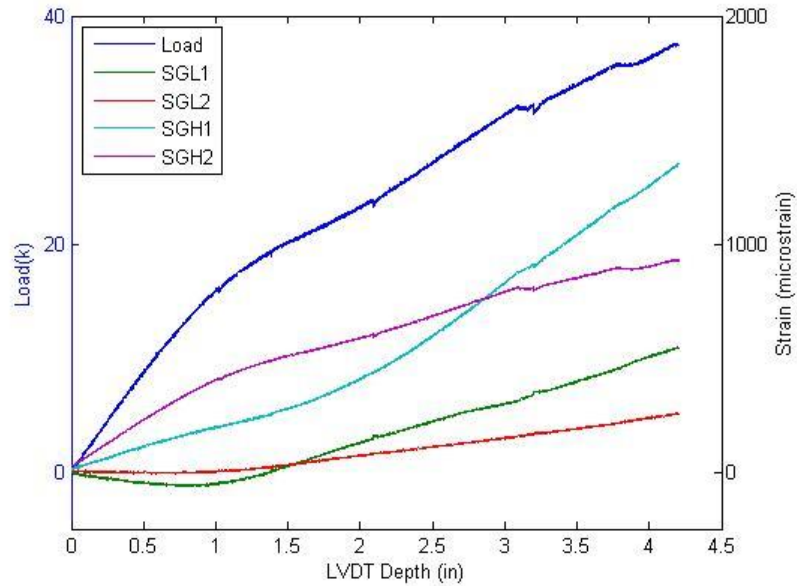


Figure 4-49: Dent Test 3 – Instrumentation

Figure 4-50 shows the cumulative signal strength for each sensor plotted against the LVDT depth. It shows the same jump in activity around 1.5" shown in Figure 4-47 and Figure 4-48. But the cumulative signal strength is dominated by the string of high energy events that began occurring around a depth 3". Note that the amplitudes plotted in Figure 4-48 are in decibels which are logarithmic. Some of these signals had amplitudes of 9V while majority of hits were below 0.5V. Since signal strength is equal to the area underneath the signal, these high amplitude events have substantially more signal strength than the lower energy events.

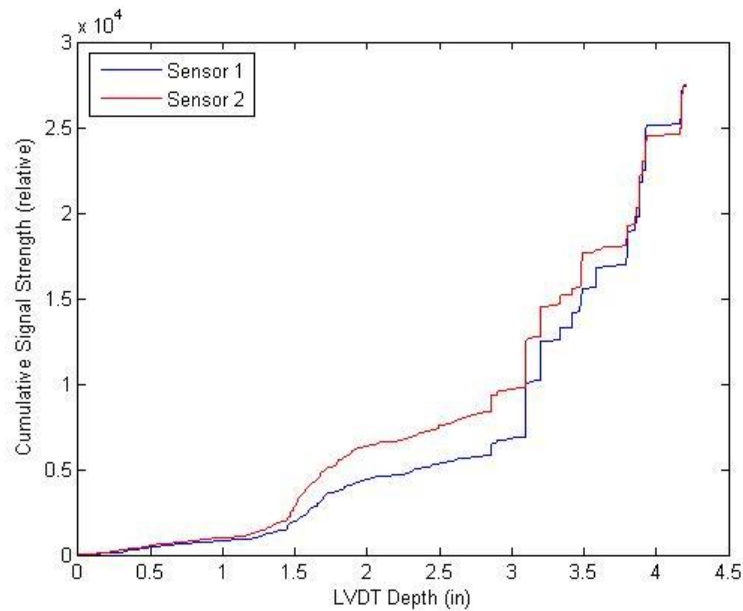


Figure 4-50: Dent Test 3 – Cumulative Signal Strength

The intensity analysis described in section 2.4.3 is applied in post processing to give a better picture of the trends shown in the cumulative signal strength plot. Figure 4-51 shows the historic index and the severity plotted individually for each sensor, and Figure 4-52 shows them plotted against each other.

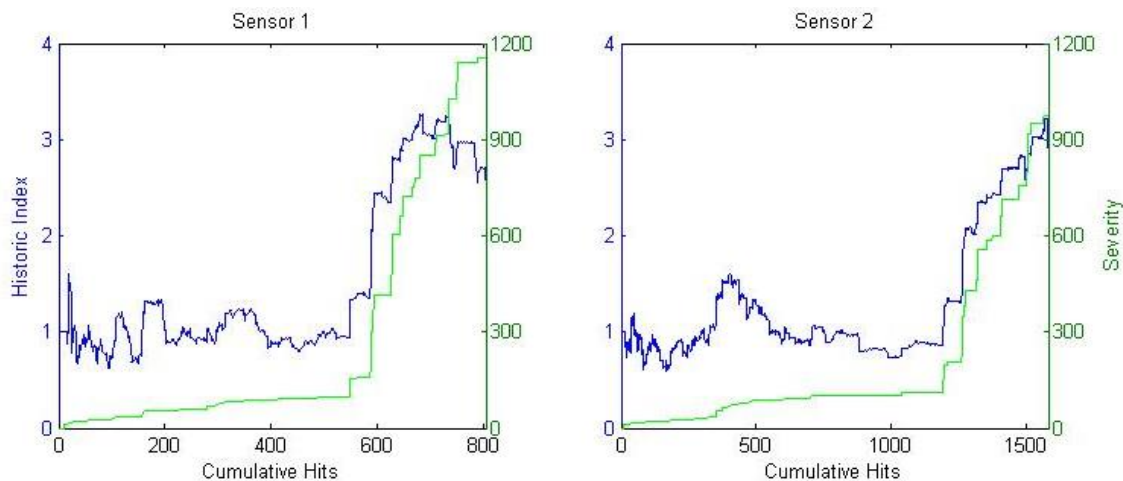


Figure 4-51: Dent Test 3 – Historic Index and Severity

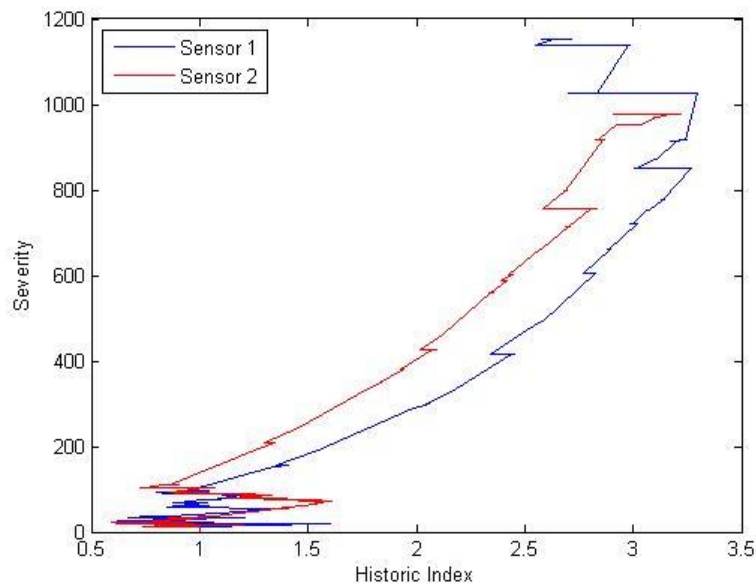


Figure 4-52: Dent Test 3 – Historic Index vs. Severity

These plots show two separate sections of the test. First, before the crack appeared and then when crack growth was relatively small, the historic index fluctuated while the severity steadily increased. Then, once the high amplitude crack growth events began occurring, both the historic index and the severity increased. This is shown by the similarities in Figure 4-51 and Figure 4-48. Additionally, it is clear that the cluster of high energy events that occurred between 1" and 2" corresponded to local increase in the historic index and a small increase in severity. These trends are shown in Figure 4-52 when the severity is relatively low but the historic index is above 1.

The following pages contain scatter plots of various parameters of individual hits obtained during the burst test. Some of them show data clusters that highlight different types of signals received during the test. With the exception of frequency based values, scatter plot parameters are normalized with 0 being the smallest value observed and 1

being the largest. It should be noted again that both sensors were on top of the pipe during this test, so these scatter plots represent many of the same events interpreted through different analog frequency filters.

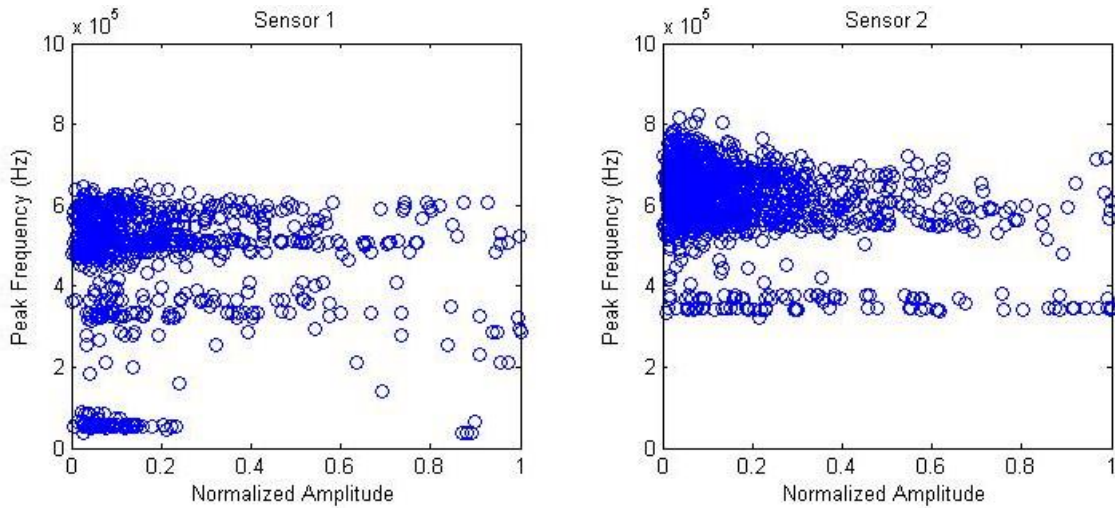


Figure 4-53: Dent Test 3 – Amplitudes vs. Peak Frequencies

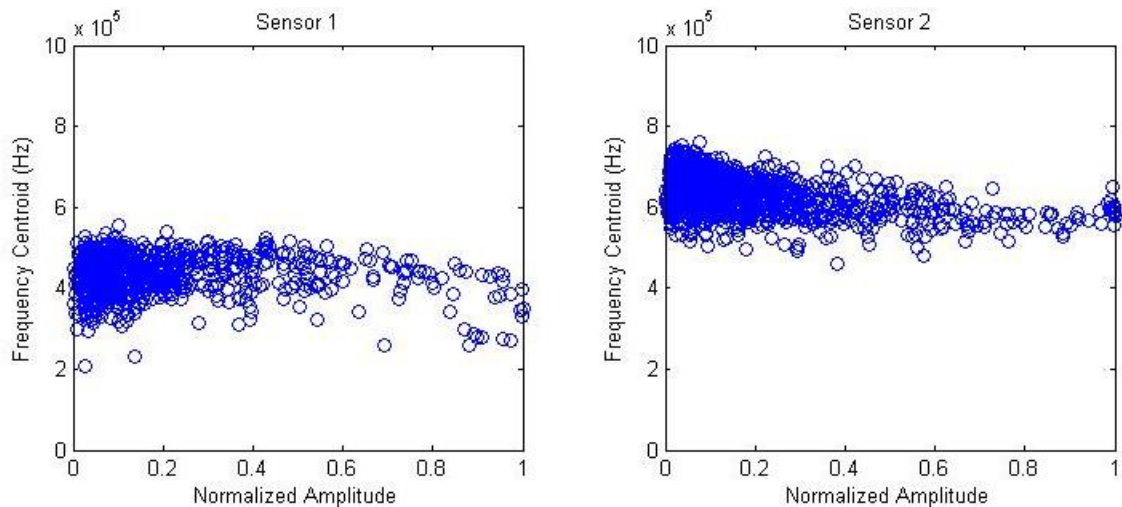


Figure 4-54: Dent Test 3 – Amplitudes vs. Frequency Centroids

Figure 4-53 and Figure 4-54 show the standard banding of frequency peaks and bunching of frequency centroids as the previous tests, with some interesting exceptions.

Sensor 1 picked up a number of hits with the dominant frequency peak below 100kHz which are indicative of some sort of background noise. Figure 4-55 shows one of these hits. The low frequency peak can be seen manifested in the waveform as a longer period oscillation carrying after the burst dissipates. The spectrum reveals that the burst contains mostly high frequency excitations, but that the low frequency peak just happens to be slightly higher than the higher peaks which defines this hit's scatter point in Figure 4-53.

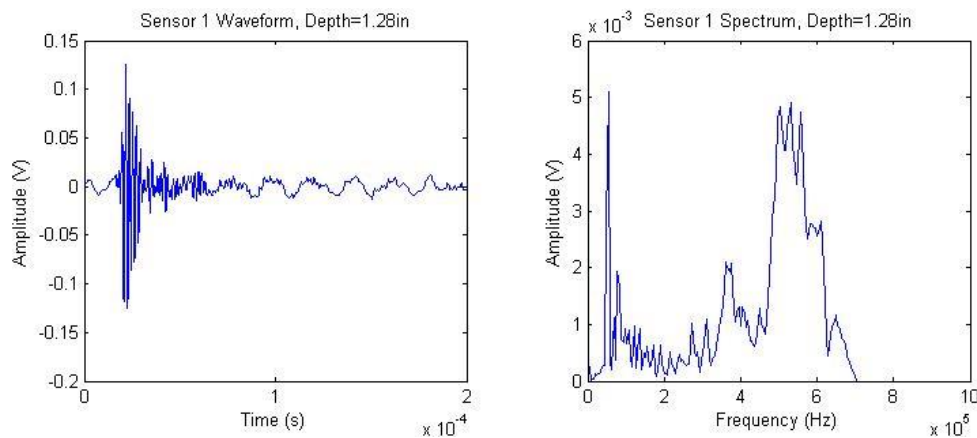


Figure 4-55: Dent Test 3 – Sensor 1 Low Frequency Peak, Weak Hit

Figure 4-53 shows that some of the low frequency peak hits were actually among the strongest received by sensor 1. These were received during the end of the test when the crack was expanding. Sensor 1 hits like this with relatively low frequency content cause the high amplitude deviation of frequency centroids shown in Figure 4-54. Figure 4-56 shows one of these strong hits and the corresponding hit from sensor 2 registered at the exact same time. This figure, along with some of the ones to follow, illustrate the differences in the waveforms interpreted through the different analog filters.

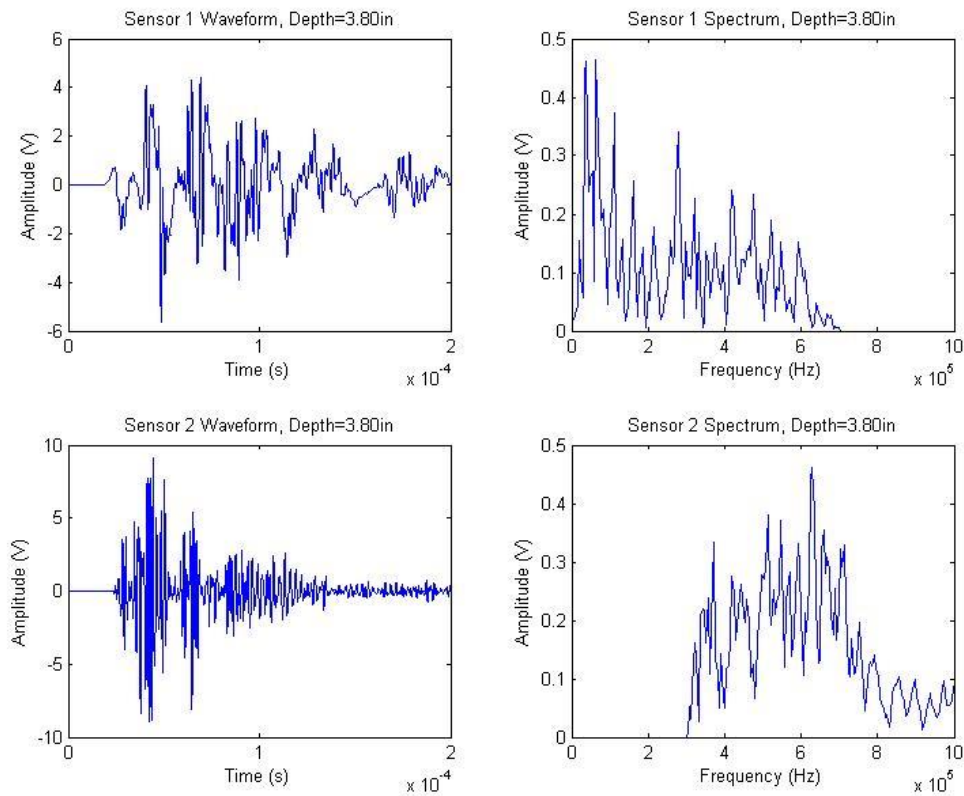


Figure 4-56: Dent Test 3 – Sensor 1 Low Frequency Peak, Strong Hit Comparison

Only one of the four high amplitude, low frequency peak hits registered by sensor 1 did not have a corresponding hit registered by sensor 2. Sensor 2, however, did record that hit's reflection 960 μ s later (compared to the 2 μ s difference in detection times in the hits shown in Figure 4-56). Figure 4-57 shows this sensor 1 hit along with the reflection picked up by sensor 2. It is unclear what prevented sensor 2 from picking up at least a weaker version of this waveform registered by sensor 1.

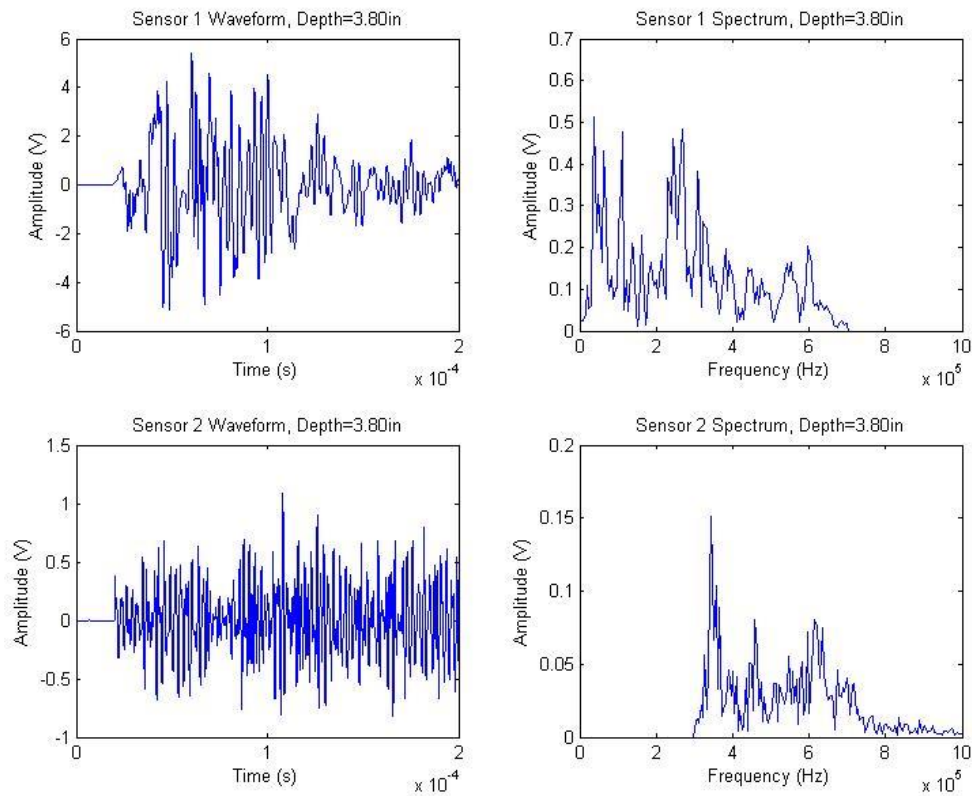


Figure 4-57: Dent Test 3 - Low Frequency Peak, Strong Hit and Reflection

Figure 4-53 also shows a distinctive band of sensor 2 signals with a frequency peak around 350kHz, which is just outside the range of the analog filter used in sensor 2. This suggests the 400-990kHz analog filter used on sensor 2 has too high of a lower limit to entirely register some of waveforms emitted by the pipes. Figure 4-58 shows one of these sensor 2 low frequency peak, high amplitude hits and the corresponding sensor 1 interpretation of the same waveform.

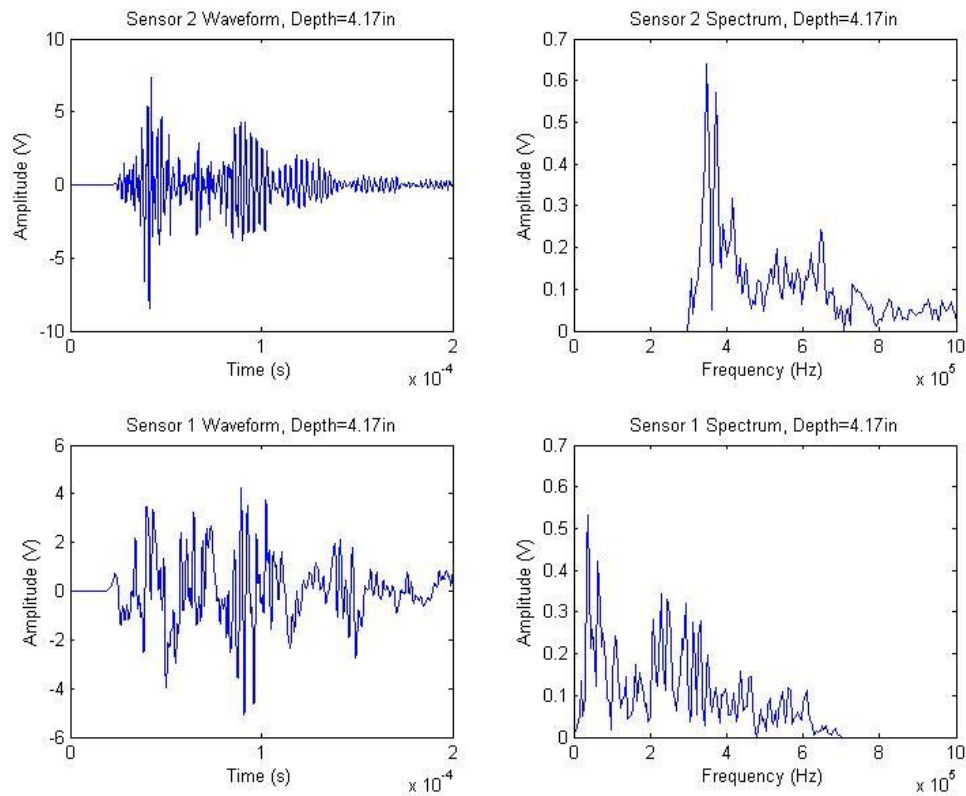


Figure 4-58: Dent Test 3 – Sensor 2 Low Frequency Peak, Strong Hit Comparison

Figure 4-59 shows a scatter plot of the depths and amplitudes of these low frequency peak hits. The sensor 1 scatter plot includes hits with peaks below 100kHz, and the sensor 2 scatter plot shows hits with peaks below 400kHz. This figure shows that the band of sensor 2 hits with low peaks shown in Figure 4-53 occurred fairly randomly throughout the test, but only after a depth of 1". Many of the strong sensor 2 hits towards the end of the test had this low frequency peak. The sensor 1 plot in Figure 4-59 shows that the noisy, low frequency peak hits described above occurred in distinctive bunches throughout the test. This indicates that there were times during the test when the

background noise was higher than normal, causing a low frequency spike in the power spectrum (as shown in Figure 4-55) in some of the signals.

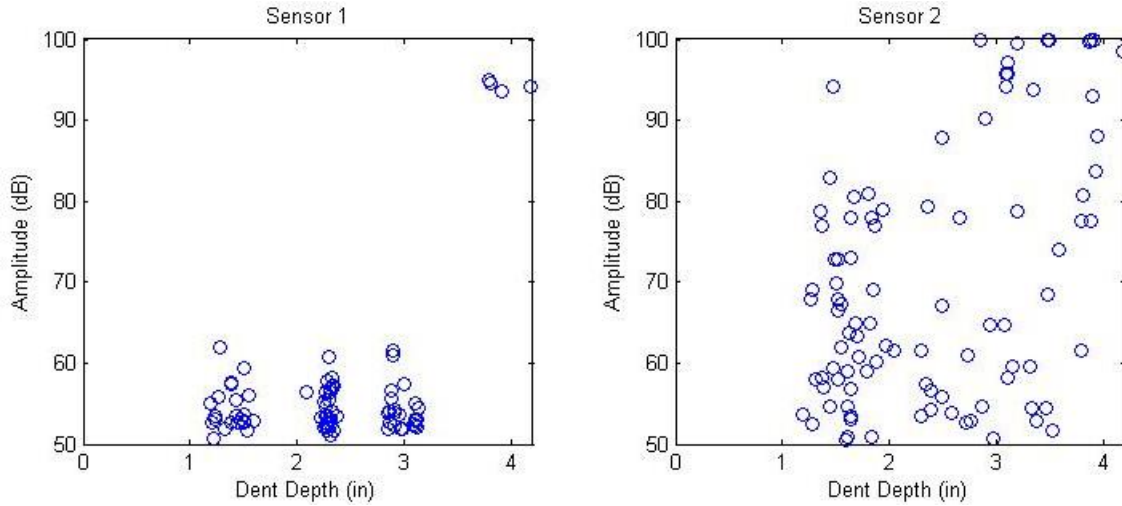


Figure 4-59: Dent Test 3 – Depths vs. Amplitudes, Low Frequency Peaks

Figure 4-60, Figure 4-61, and Figure 4-62 show scatter plots representing the durations and rise times of the signals from each sensor. Figure 4-63 shows a scatter plot of depths and durations.

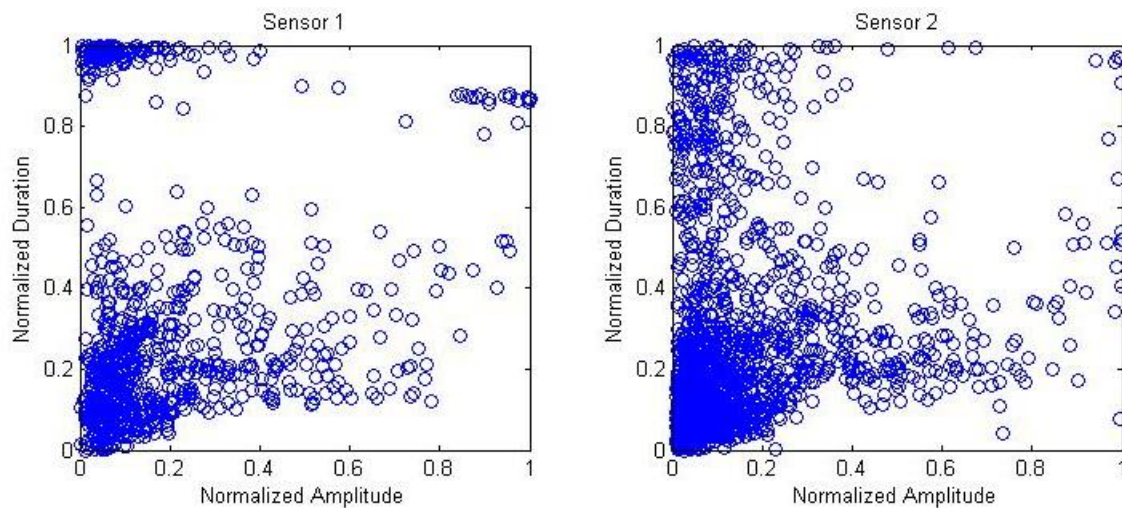


Figure 4-60: Dent Test 3 – Amplitudes vs. Durations

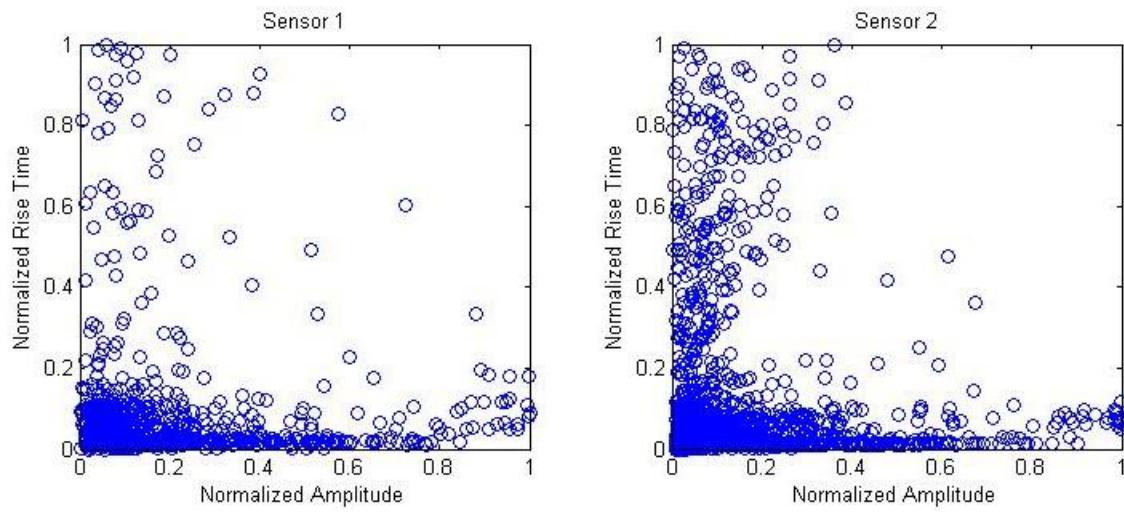


Figure 4-61: Dent Test 3 – Amplitudes vs. Rise Times

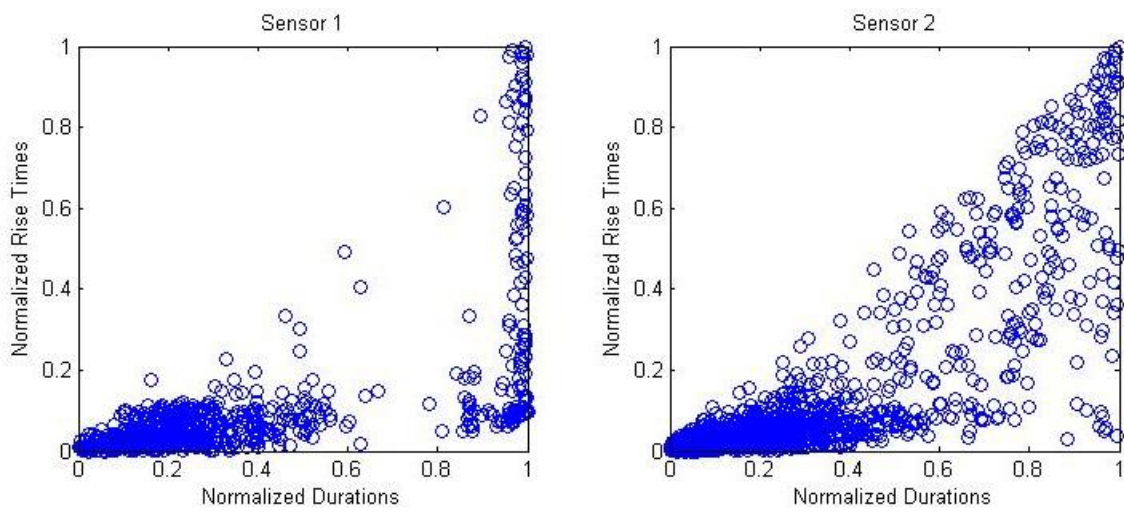


Figure 4-62: Dent Test 3 – Durations vs. Rise Times

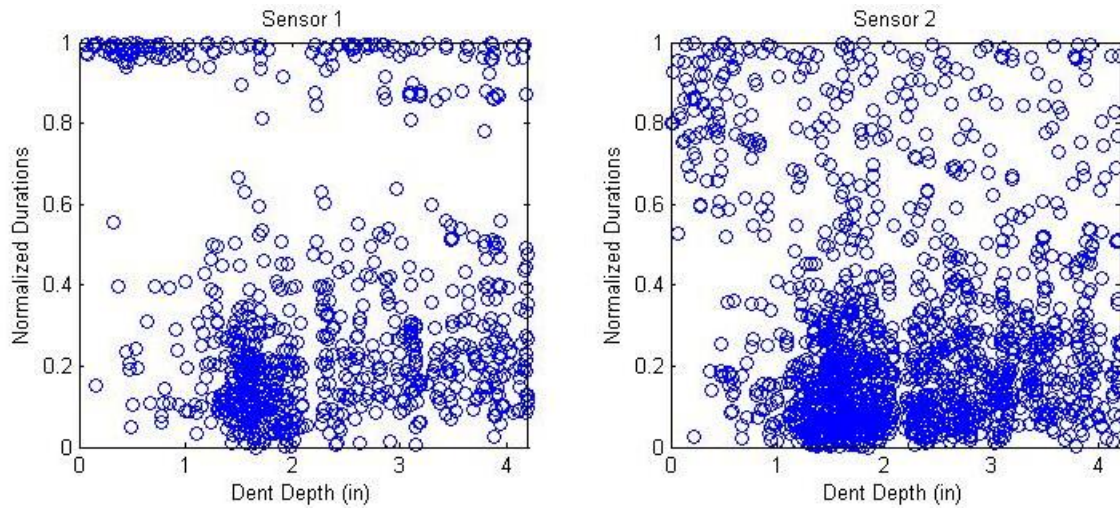


Figure 4-63: Dent Test 3 – Depths vs. Durations

These scatter plots—Figure 4-62 in particular—show that sensor 2 registered a lot more hits with medium durations and rise times than sensor 1. Earlier analysis indicates that these are indications of a hit occurring farther away from the sensor. After separating out this cluster of hits in post-processing, it appears that this difference in signals is not a result of the differences in frequency filters, rather the differences in software thresholds used during the test.

Sensor 1 was set at a 50db threshold because of a strong, low frequency background noise, and sensor 2 was set at a 35db threshold because there was very little high frequency noise. Even though all threshold based parameters like duration and counts were calculated using the floating post-processing threshold described in section 3.1.2, the waveforms were still generated and saved by the software using the original thresholds. The definition time of the first count can have a large impact on the signal

generated for more spread out waveforms because the software only saves 200 μ s for each signal.

Figure 4-64 shows one of these long rise time hits registered by sensor 2, along with thresholds of 35db (the sensor 2 software threshold), 40db (the post-processing threshold), and 50db (the sensor 1 software threshold). The signal has been zoomed on the right so that the first count at each threshold can be seen.

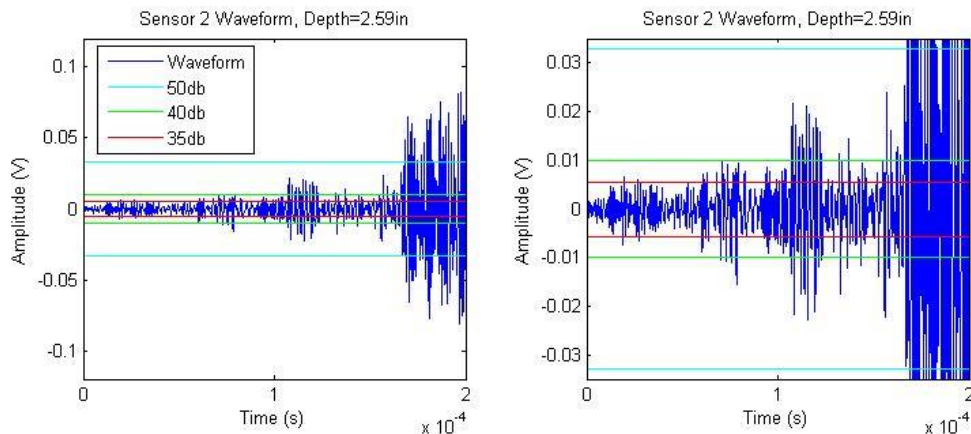


Figure 4-64: Dent Test 3 – Sensor 2 Long Rise Time Hit

The software generated the waveform above by saving a 200 μ s signal beginning 25 μ s before it first crossed the 35db software threshold used. Post processing defined its rise time and duration using the 40db threshold, which it crossed about 50 μ s after the 35db threshold. The signal was not stronger than 50db until near the end of the defined waveform. If the software generated this waveform beginning at 25 μ s before it surpassed 50db, the signal would have done a much better job of representing the actual mechanical waveform travelling through the pipe.

Closer examination of Figure 4-62 reveals that sensor 1 and sensor 2 both have plenty of signals with large durations, but sensor 2 has a more spread out distribution of

rise times because of how it defined the beginning of signals. Sensor 1 shows a cluster of signals with the maximum duration allowed by the software, but as explained in section 4.2 the true durations of these hits are likely longer than the 200 μ s waveforms generated and the true scatter plot would be more spread out.

Figure 4-65 shows another sensor 2 waveform interpreted by the post-processing to have a long rise time. However, the signal clearly captured two waveforms that occurred at almost the same time. These dual hits were rare but more common at sensor 2 than sensor 1.

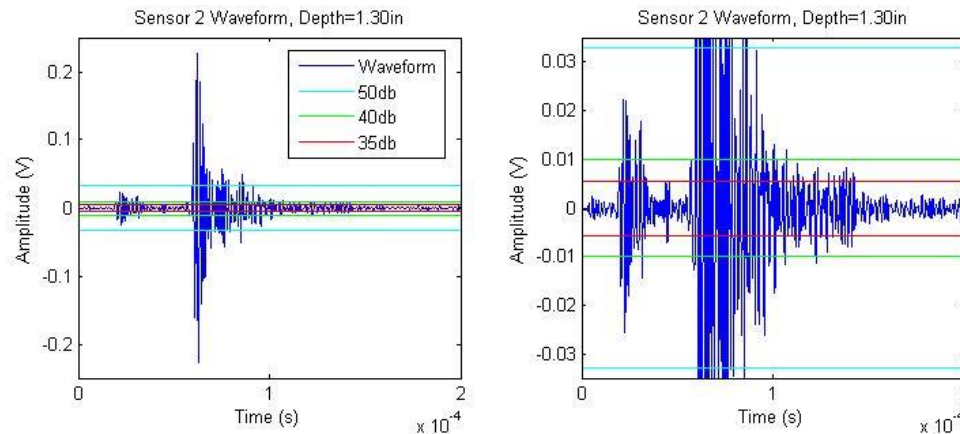


Figure 4-65: Dent Test 3 – Sensor 2 Dual Hit

Post-processing calculates the rise time as the time between when the signal crosses 10mV (or 5% peak amplitude for stronger hits) and the time of the peak amplitude. Since the peak amplitude of this signal is 226mV, the rise time of this hit was actually calculated using a threshold of 11.3mV which is very close to 40db. But the point is that since this waveform actually shows a weaker hit immediately followed by a stronger it, post-processing misinterpreted its duration and rise time. Had this signal

been generated with a 50db software threshold, the weaker hit would have been ignored and the software would have generated a waveform based only on the larger hit.

Figure 4-66 shows another dual hit where unfortunately a relatively small event occurred just 150 μ s before one of the 30 largest hits detected during Dent Test 3. In this case the small hit was around 55db, so even if the sensor 2 threshold was set at 50db this misinterpretation would have still been captured. But the 55db hit is so small compared to the 99.7db hit that it cannot even be seen in the first half of Figure 4-66.

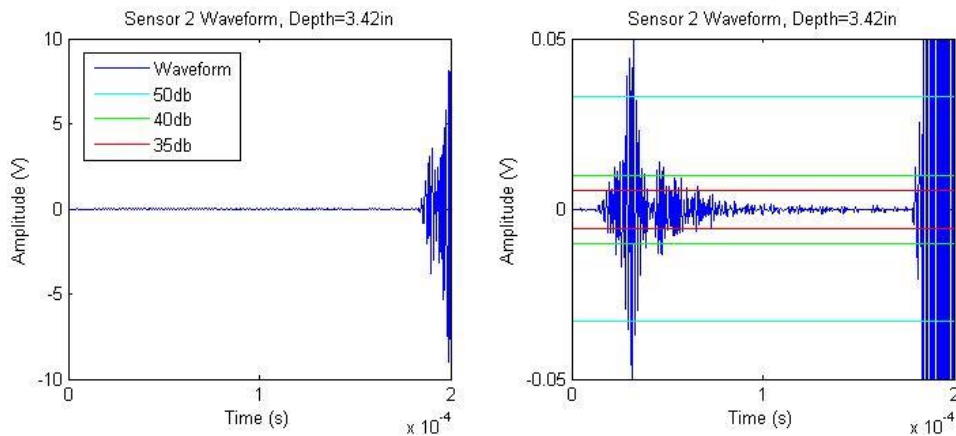


Figure 4-66: Dent Test 3 – Sensor 2 Large Dual Hit

Figure 4-67 compares this same signal with the corresponding signal generated by sensor 1 at the same time representing the same event. Sensor 1 did not pick up the dual signal: the smaller event must have been below 50db when interpreted by the 100-600kHz filter. Figure 4-67 shows that this dual hit phenomenon not only skews the time based parameters like rise time and duration, but since the software was not able to capture the entire strong hit, the frequency spectrum is not nearly as well defined as it was for sensor 1's signal.

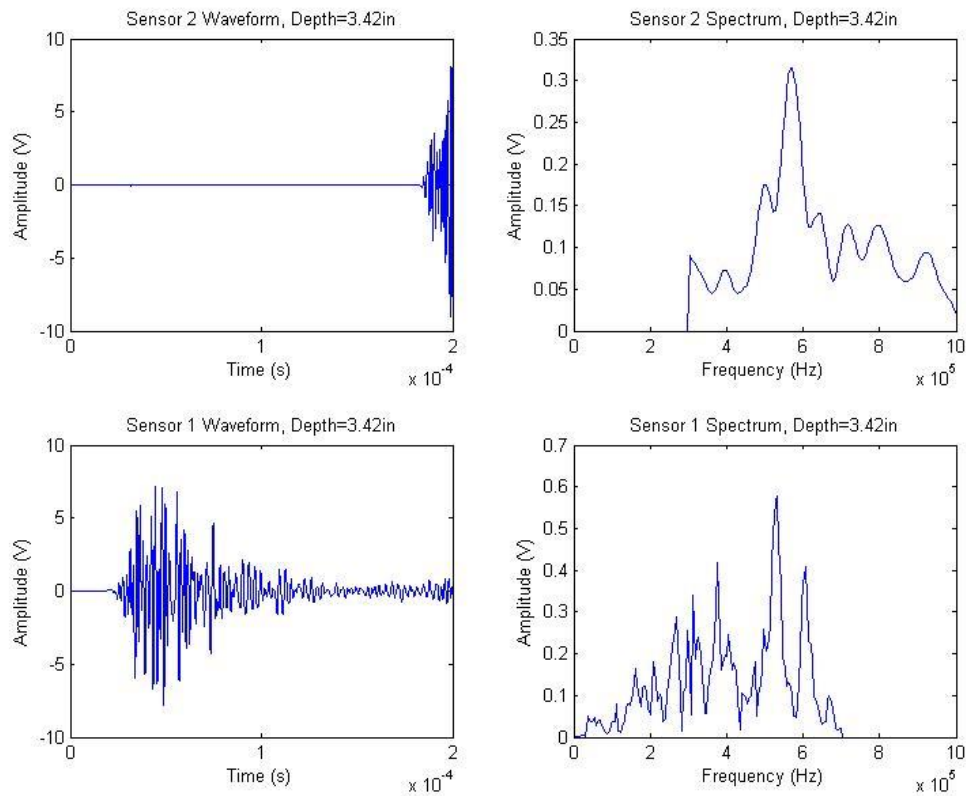


Figure 4-67: Dent Test 3 – Sensor 2 Large Dual Hit Comparison

These dual hits—misinterpreted as single hits—suggest the possibility of another potential malfunction of the AEwin™ software. As described in section 3.1, the AEwin™ software uses a HLT (hit lockout time) of 1000μs to prevent reflections from being registered as separate hits. It seems that the user is not allowed to change this HLT. These hits show that sometimes events occur within 200μs of each other. It must be even more likely for hits to occur within 1000μs of each other, which suggests that sometimes the software has the unwanted effect of ignoring an event if it occurred before the HLT from a previous event was over. Had the 55db preliminary hit shown in Figure 4-66 occurred just 100μs earlier, sensor 2 would have entirely missed registering

one of the largest events that occurred during the test. Ironically, it would have probably recorded its reflection after the waveform travelled around the pipe, since that would have arrived after the HLT was over.

This event lockout problem is more likely to occur at lower thresholds because weaker events are usually more common than stronger hits. While the threshold for sensor 2 was set as low as possible in an attempt to capture as many events as possible, this low threshold might have had the unwanted effect of causing weak events (which ended up being ignored in post-processing anyways) to lockout stronger events.

Sensor 2 initially registered 21407 hits during the approximately 7500 seconds that the actuator was pushing Dent Pipe 3. Since each hit caused a lockout time of 1000 μ s, this means that sensor 2 was locked out for 21.4 seconds over the course of the test (0.29% of the test). Sensor 1, which only registered 938 hits because of its higher threshold, was locked out for less than a second of testing. Since 1582 sensor 2 hits were above 50db which is one every 5 seconds on average, so this hints at the fact that sensor 2 did lockout a few relatively strong hits. Since Figure 4-48 shows that high amplitude events sometimes occurred in bunches, the true number of missing hits is probably larger than this quick math suggests.

These dents were created relatively slowly (0.034in/min) so the acoustic activity was spaced out. Had the dents been created at 1.0in/min, the test would have only taken 255 seconds and sensor 2 would have been locked out for up to 8.5% of that time. This analysis suggests that low thresholds are dangerous—especially for fast tests—when the primary goal is the recording of high energy activity.

Despite the shortcomings in the definitions of durations and rise times, k-means clustering as described in section 2.4.4 was able to properly separate events into different groups based on their parameters. A k-means algorithm with three cluster centers separated the Dent Test 3 events based on amplitude, duration, and rise time, and the results naturally recreate some of the proposed differentiations described above. The figures below show the same scatter plots as above, with each event colored with its corresponding k-means cluster.

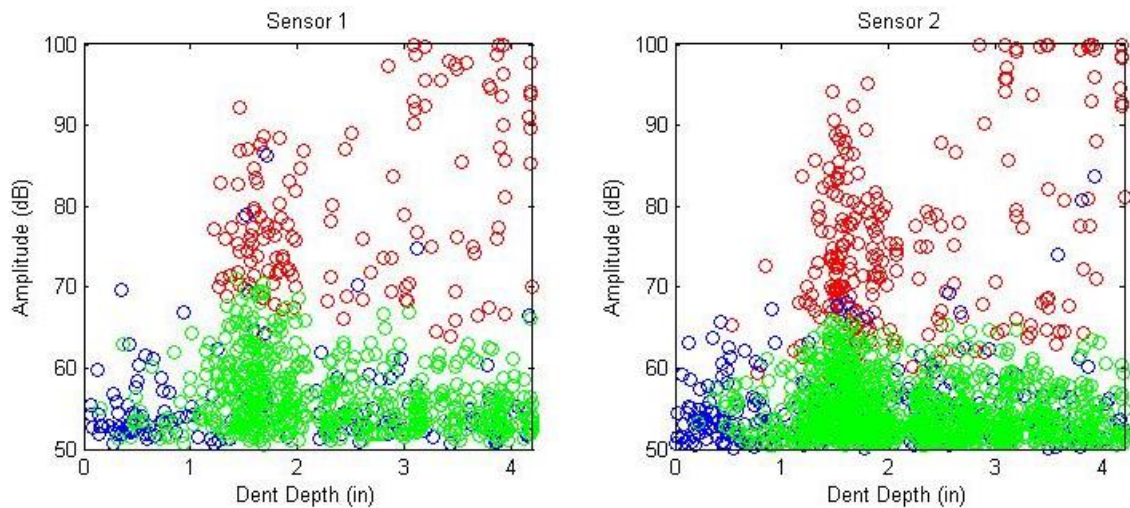


Figure 4-68: Dent Test 3 – K-Means Depths vs. Amplitudes

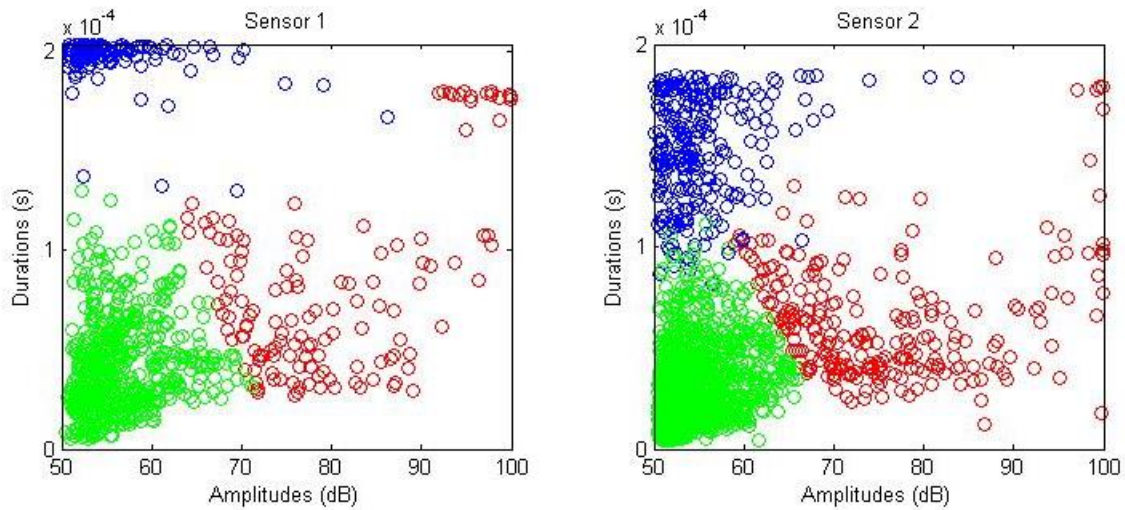


Figure 4-69: Dent Test 3 – K-Means Amplitudes vs. Durations

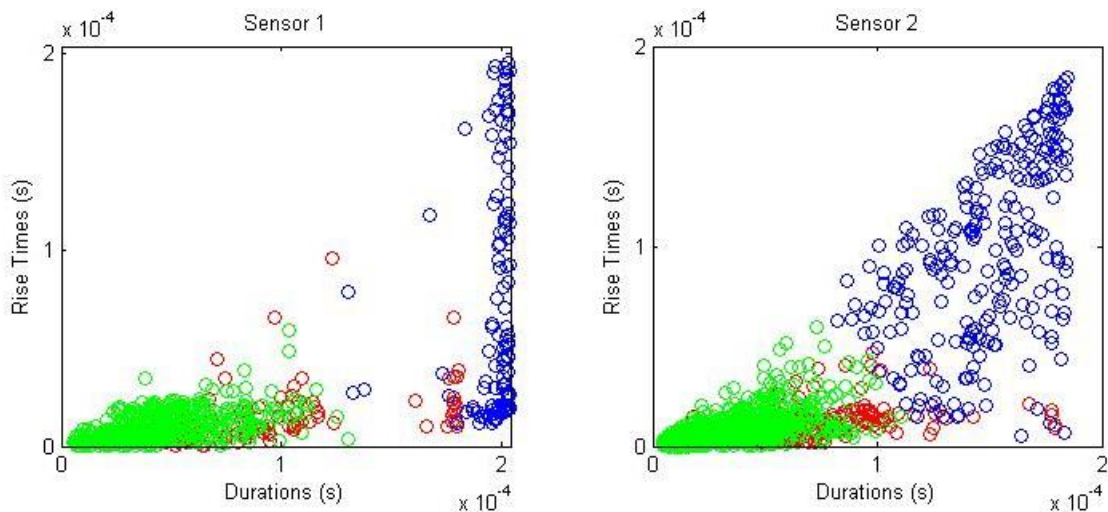


Figure 4-70: Dent Test 3 – K-Means Durations vs. Rise Times

The figures above show that the algorithm successfully separated the hits into long duration hits, weak short duration hits, and strong short duration hits. Figure 4-68 shows that the hits that were are most likely associated with crack initiation and growth were grouped into the red cluster. Since the blue hits shown above generally represent

deformation events that occurred farther away from the sensor (and thus the dent), they could potentially be separated out and ignored for future analyses.

CHAPTER V

SUMMARY AND CONCLUSIONS

5.1 DENT TEST CONCLUSIONS

It is difficult to make significant conclusions from the AE data measured during the dent tests. This is because each individual dent test was under different circumstances, both in regards to the test setups and the quality of the AE data. That being said, there is a lot of interesting trends shown in the data above. These are some of the potential conclusions regarding the characterization of the acoustic activity measured during the pipe dent tests.

1. Dent Test 1 suggests that the detection of “sharp” hits, defined as hits with relatively short rise times and large amplitudes, indicates that deformation events are occurring near the location of the sensor. The similarities between the distribution of sharp hits at sensor 1 and the strains in SGL1 (which were placed in identical spots on opposite sides of the pipe) show that strain levels are closely connected to the accumulation rates of sharp hits.
2. Similarly, comparison of the hits obtained by different sensors in Dent Tests 1 and Dent Tests 2 show that acoustic signals with long durations and rise times tend to be generated by events that occurred farther away from the sensor.

3. The peculiar findings of Dent Test 2 and the API X52 tensile tests suggest that certain pipe steels may not be as acoustically active as others.
4. Based on the lack of AE activity during the necking stages of tensile tests, it may not be possible to use tensile samples to aid in the characterization of acoustic emissions generated at the pipes' critical strain. But this does not necessarily mean that the characterization is impossible. AE activity is a function not only of the source event, but of the material geometry and the method of loading. Self-characterization established over more dent tests may reveal patterns which are able to reveal the acoustic activity at critical strain.
5. Dent Test 3 offers a great example of the kinds of acoustic activity associated with crack generation and growth in the pipe dent tests. High energy, sharp hits significantly increased around the time that the crack was observed. Even higher energy hits were observed during the later stages of crack growth. These high energy ultrasonic emissions were directly accompanied by the audible metallic "pings" sometimes heard during dent tests.
6. The data clustering techniques applied to Dent Test 3 were able to separate the acoustic events which seem to be associated with crack development.

5.2 ACOUSTIC EMISSIONS CONCLUSIONS AND RECOMMENDATIONS

A number of lessons were learned about the nature and behavior of the AE monitoring system used during this test. What follows is a summary of recommendations for future acoustic emissions testing based on the findings in this study.

1. The quality of sensor-to-surface connectivity has a large impact on the reliability of the signals generated by the transducers. The surface underneath the transducer tip should be polished, and proper couplant (roughly a pea sized drop) must be applied. Additionally, the method by which the transducers are mounted must allow them to maintain a proper, flush connection throughout the test.
2. PLB calibration tests should always be conducted after sensors are set in place and before testing begins. A PLB test (as described in section 3.1.1) generated one inch away from the transducer tip should generate a signal with an amplitude of 75-80dB. If the signal is too weak, the surface may need to be cleaned more, couplant may need to be reapplied, and/or the sensor may not be making appropriate contact with the surface.
3. The sensor-to-source distance also has a large impact on the quality of generated signals. The transducer should be placed as close as possible to the expected location of the majority of deformation.
4. The software analog filter determines many of the characteristics of the generated signals. Further research may be needed to determine exactly

which frequency bands are best to observe for specific materials.

Consistency in filter selection between tests is important.

5. The software threshold is important for a number of reasons. Before any test, after calibration tests are successful, the threshold can be lowered and the background noise level can be observed. The threshold should be set above the measured ambient background noise level to prevent the sensor from flooding with signals. However, the lower the threshold the earlier the definition of the beginning of the waveform. It is more likely for longer waveforms (generated further from the sensor) to be cut off too early with a low threshold. Also, the lower the threshold the more likely it is for a signal to be blocked by the HLT of an earlier signal. As long as it sits above noise levels, thresholds between 40dB and 50dB should be adequate.
6. Unfortunately, the post-processing capabilities of the AEWin™ Software are limited, and importing and analyzing all of the data in MATLAB® is quite complicated. But the AEWin™ Software is able to apply data file filters and create an ASCII Output which is a text file containing information about each hit generated during a test. This output may be adequate information for most acoustic emissions applications. However, if deeper research is to be conducted, MATLAB® provides much more versatile post-processing abilities. The MATLAB® scripts and functions used for this study are included in this appendix and are free to be copied, adjusted, and applied by someone with appropriate knowledge of MATLAB®.

5.3 DIRECTIONS FOR FUTURE RESEARCH

Since this study was the first attempt at applying an acoustic emissions monitoring system to the critical strain research program, there are plenty of directions to proceed.

1. Because of the limitations of the AEWin[™] Software, it may be better to eventually develop a LabVIEW program so that the data obtained by the transducers and the DAQs can be more seamlessly integrated with other instrumentation data. This may also be able to alleviate some of the issues with the AEWin[™] signals, like the 200 μ s limit on waveform size.
2. The transducers used in this research are not ideal for the pipe dent tests. As the CTS Valpey Corporation website states, they are intended for blast and shock wave measurements. A more specialized transducer with a specific frequency band may be more capable of producing the desired results in the pipe dent tests. MISTRAS (Physical Acoustics), the company that manufactures the DAQs used in this research, produces a variety of transducers of different sizes and spectra. A sensor with a frequency band around 300-700kHz should be adequate for the API X65 pipes.
3. Acoustic emissions monitoring of the pipe dent tests should continue so that more can be learned about the nature of the AE data obtained during these studies.

4. A better way to apply the transducers to tensile tests may need to be established due to the current variability in data quality.
5. More research is needed to determine why the API X52 pipe and tensile specimen exhibited such quiet behavior. This may be as simple as finding a transducer with the right frequency band.
6. The AE monitoring system should be applied to other tests carried out in the laboratory. The best way to learn more about the capabilities of this technology is to apply to a variety of scenarios to figure out what works and what doesn't.

REFERENCES

1. Miller, R. K., Hill, E. v.K., and Moore, P. O. *Nondestructive Testing Handbook Volume 6: Acoustic Emission Testing*, 3rd Edition. American Society for Nondestructive Testing (2005).
2. *Standard Terminology for Nondestructive Examinations*. American Society for Testing of Materials, E1316. ASTM International (2014).
3. Kalyanasundaram, P., Mukhopadhyay, C.K., and Subba Rao, S.V. *Practical Acoustic Emission*. Alpha Science International Ltd (2007).
4. Shi, Z., Jarzynski, J., Hurlebaus, S., and Jacobs, L.J. *Characterization of Acoustic Emission Signals from Fatigue Fracture*. Proceedings of the Institution of Mechanical Engineers, Vol 214 Part C (2000).
5. Jacobs, L.J., Scott, W.R., Granata, D.M., and Ryan, M.J. *Experimental and Analytical Characterization of Acoustic Emission Signals*, Journal of Nondestructive Evaluation, Vol. 10 p. 63-70 (1991).
6. Hurlebaus, S., Jacobs, L.J., and Jarzynski, J. *Laser Techniques to Characterize the Effect of Geometry on Acoustic Emission Signals*. Nondestructive Testing and Evaluation, 14:1-2 p. 21-37 (1998).
7. Nesvijski, E., and Marasteanu, M. *Wavelet Transform and its Applications to Acoustic Emission Analysis of Asphalt Cold Cracking*. NDT.net (2007).
8. Kaiser, J. Dissertation, Technische Hochschule Munchen, Germany (1950).
9. Callister, W.D. *Materials Science and Engineering*, 7th Edition. John Wiley & Sons, Inc. (2007).
10. Cone, D.M. Thesis, The University of Texas at El Paso, TX (2006).

11. Ould Amer, A., Gloanec, A-L., Courtin, S., and Touze, C. *Characterization of Fatigue Damage in 304L Steel by an Acoustic Emission Method*. Procedia Engineering, 66 p. 651-660 (2013).
12. Por, G., Doszpod, L., Vekony, A., Danko, Z., and Gonda, V. *Acoustic Emission during Tensile Tests and ECAP*. 30th European Conference on Acoustic Emission Testing & 7th International Conference on Acoustic Emission, University of Granada (2012).
13. Marsudi, M. *Study of Acoustic Emission During Tensile Test of Mild Steel Plate*. 8th International Conference on Heat Transfer, Fluid Mechanics and Thermodynamics (2011).
14. Bagherifaez, M., Behnia, A., Majeed, A.A., and Kain, C.H. *Acoustic Emission Monitoring of Multicell Reinforced Concrete Box Girders Subjected to Torsion*. The Scientific World Journal, ID 567619 (2014).
15. Degala, S. Thesis, University of Pittsburgh, PA (2008).
16. Godin, N., Huguet, S., Gaertner, R., and Salmon, L. *Clustering of Acoustic Emission Signals Collected during Tensile Tests on Unidirectional Glass/Polyester Composite using Supervised and Unsupervised Classifiers*. NDT&E International, 37 p. 253-264 (2004).
17. Gostautas, R.S. *Identification of Failure Prediction Criteria using Acoustic Emission Monitoring and Analysis of GFRP Bridge Deck Panels*. Kansas Department of Transportation (2007).
18. Nor, N.M., Bunnori, N.M., Ibrahim, A., Shahidan, S., and Saliah, S.N.M. *Relationship between Acoustic Emission Signal Strength and Damage Evaluation of Reinforced Concrete Structure: Case Studies*. IEEE Symposium on Industrial Electronics and Applications (2011).
19. Abdelrahman, M. Thesis, University of South Carolina, Columbia, SC (2013).

20. Hossain, M. Dissertation, University of South Carolina, Columbia, SC (2013).
21. Keshtgar, A., and Modarres, M. *Detecting Crack Initiation Based on Acoustic Emission*. Chemical Engineering Transactions, Vol. 33 p. 547-552 (2013).
22. Marfo, A., Chen, Z., and Li, J. *Acoustic Emission Analysis of Fatigue Crack Growth in Steel Structures*. Journal of Civil Engineering and Construction Technology, Vol. 4(7) p. 239-249 (2013).
23. Schultz, A. *Acoustic Emission Monitoring of a Fracture-Critical Bridge*. Minnesota Department of Transportation (2014).
24. Shaira, M., Godin, N., Guy, P., Vanel, L., and Courbon, J. *Evaluation of the Strain-Induced Martensitic Transformation by Acoustic Emission Monitoring in 304L Austenitic Stainless Steel: Identification of the AE Signature of the Martensitic Transformation and Power-Law Statistics*. Materials Science and Engineering A, 492 p. 392-399 (2008).
25. Yu, J., Ziehl, P., Matta, F., and Pollock, A. *Acoustic Emission Detection of Fatigue Damage in Cruciform Welded Joints*. Journal of Constructional Steel Research, 86 p. 85-91 (2013).
26. Ai, Q., Liu, C-X., Chen, X-R., He, P., and Wang, Y. *Acoustic Emission of Fatigue Crack in Pressure Pipe under Cyclic Pressure*. Nuclear Engineering and Design, 240 p. 3616-3620 (2010).
27. Budano, S., Giunta, G., Lucci, A. *Acoustic Emission Data Analysis to Evaluate Damage Mechanisms in Pipeline Carbon Steels*. NDT.net (2011).
28. De la Rosa, J.J.G., Piotrkowski, R., Ruzzante, J. *Third-Order Spectral Characterization of Acoustic Emission Signals in Ring-Type Samples from Steel Pipes for the Oil Industry*. Mechanical Systems and Signal Processing, 21 p. 1917-1926 (2007).

29. Shehadeh, M., Steel, J.A., and Reuben, R.L. *Acoustic Emission Source Location for Steel Pipe and Pipeline Applications: The Role of Arrival Time Estimation*. Proceedings of the Institution of Mechanical Engineers, Vol 220 Part E (2006).
30. Barat, K., Bar, H.N., Mandal, D., Roy, H., Sivaprasad, S., and Tarafder, S. *Low Temperature Tensile Deformation and Acoustic Emission Signal Characteristics of AISI 304LN Stainless Steel*. Materials Science & Engineering A, 597 p. 37-45 (2014).
31. Kotoul, M. and Bilek, Z. *Acoustic Emission During Deformation and Crack Loading in Structural Steels*. International Journal of Pressure Vessels & Piping, 44 p. 291-307 (1990).
32. Sagar, R.V. *An Experimental Study on Acoustic Emission Energy and Fracture Energy of Concrete*. Proceedings of the National Seminar & Exhibition on Non-Destructive Evaluation.
33. Fowler, T.J., Blessing, J.A., and Strauser, F.E. *Intensity Analysis*. Proceedings of the 4th International Symposium on Acoustic Emission from Composite Materials, p. 16-27 (1992).
34. Panjsetooni, A., Bunnori, N.M., Shirkhani, Ze., Shirkhani, Zo. *Evaluation of Welded Steel Beam Using Acoustic Emission*. Caspian Journal of Applied Sciences Research, 2(6) p. 43-50 (2013).
35. Grosse, C. *Hsu-Nielsen Source*. NDT.net (2002).
36. Sause, M.G.R. *Investigation of Pencil-Lead Breaks as Acoustic Emission Sources*. Journal of Acoustic Emission, 29 p. 184-196 (2011).
37. GE Infrastructure. *Sound Speeds and Pipe Size Data*. Installation Reference 914-004E1 (2014)

APPENDIX A

MATLAB® CODES

A.1 – AEAnalysis.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% AE ANALYSIS SCRIPT %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% By Jeffrey Michel
% Texas A&M University
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This is the master script for the majority of the analyses performed
% in this paper. By switching the variable "whichtest" based on the
% function SetTest.m, each of the pipe and tensile tests could be
% analyzed individually without the need for their own separate
% scripts. Specific sections of this code could be copied and run based
% on which analyses were performed, along with a number of pre-written
% filters and graphs.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear all;
close all;
clc;

% Switch between floating and fixed thresholds.
threshtype = 'floating';
% threshtype = 'fixed';

%% TEST SELECTION
% Pick test & generate AE data structure.

whichtest = 'DentTest3-50db';
[Idata,testparam] = SetTest(whichtest);
AEdata = CreateAEdata(testparam,threshtype);
backup1 = AEdata;

%% PLB TESTS
% Some tests had PLB tests within the main files that had to be
removed.

for i = 1;
    if strcmp(whichtest,'DentTest3') == 1
        % PLB 1
        AEdata(1).event(101) = [];
        AEdata(2).event(1498) = [];
        % PLB 2
        AEdata(1).event(474) = [];
        AEdata(1).event(474) = [];
        AEdata(2).event(5140) = [];
        AEdata(2).event(5140) = [];
        % PLB 3
        AEdata(1).event(735) = [];
        AEdata(1).event(735) = [];
        AEdata(1).event(735) = [];
```

```

        AEdata(1).event(735) = [];
        AEdata(1).event(735) = [];
    end
    if strcmp(whichtest, 'DentTest3-50db') == 1
        % PLB 1
        AEdata(1).event(81) = [];
        AEdata(2).event(176) = [];
        % PLB 2
        AEdata(1).event(433) = [];
        AEdata(1).event(433) = [];
        AEdata(2).event(953) = [];
        AEdata(2).event(953) = [];
        % PLB 3
        AEdata(1).event(661) = [];
        AEdata(1).event(661) = [];
        AEdata(1).event(661) = [];
        AEdata(1).event(661) = [];
        AEdata(1).event(661) = [];
    end
end

%% DATA FILTER
% Run individual filters, or make new ones by copying and switching if
% statement.

% Counts of 3 or less
for sn = 1:length(AEdata)
    fin = length(AEdata(sn).event);
    for en = 1:fin
        disp(sprintf('%d %d', sn, en))
        if AEdata(sn).event(fin+1-en).counts <= 3
            AEdata(sn).event(fin+1-en) = [];
        end
    end
end

% Low Frequency Centroid
for sn = 1:length(AEdata)
    fin = length(AEdata(sn).event);
    for en = 1:fin
        if AEdata(sn).event(fin+1-en).freqcent <= 2e5
            AEdata(sn).event(fin+1-en) = [];
        end
    end
end

% For Dent Test 3, below 50db
for sn = 1:length(AEdata)
    fin = length(AEdata(sn).event);
    for en = 1:fin
        if AEdata(sn).event(fin+1-en).decamp <= 50
            AEdata(sn).event(fin+1-en) = [];
        end
    end
end

```

```

        end
    end

    % % No counts after peak
    % for sn = 1:length(AEdata)
    %     fin = length(AEdata(sn).event);
    %     for en = 1:fin
    %         if AEdata(sn).event(fin+1-en).countsafterpeak == 0;
    %             AEdata(sn).event(fin+1-en) = [];
    %         end
    %     end
    % end
    %
    % % Rise Time = 0
    % for sn = 1:length(AEdata)
    %     fin = length(AEdata(sn).event);
    %     for en = 1:fin
    %         if AEdata(sn).event(fin+1-en).risetime == 0;
    %             AEdata(sn).event(fin+1-en) = [];
    %         end
    %     end
    % end
    %
    % % Very high amplitude for Burst Test
    % for sn = 1:length(AEdata)
    %     fin = length(AEdata(sn).event);
    %     for en = 1:fin
    %         if AEdata(sn).event(fin+1-en).amplitude >= 7;
    %             AEdata(sn).event(fin+1-en) = [];
    %         end
    %     end
    % end

%% SENSOR LOCKOUT
% For tests in which the sensors were placed on opposite sides of the
pipe,
% this code determines which events were picked up by both sensors and
% removes the signal for the sensor that picked it up later. These
% duplicates are stored for further analysis.

S=0;
for i = 1:length(AEdata(1).event)
    disp(sprintf('%d',i))
    if i == length(AEdata(1).event)
        break
    end
    for j = 1:length(AEdata(2).event)
        if j == length(AEdata(2).event)
            break
        end
        if abs(AEdata(2).event(j).at - AEdata(1).event(i).at) <= 0.003
            if AEdata(2).event(j).at < AEdata(1).event(i).at

```

```

        S=S+1;
        lockouthits(S) = AEdata(1).event(i);
        AEdata(1).event(i) = [];
        i=i-1;
    elseif AEdata(1).event(i).at <= AEdata(2).event(j).at
        S=S+1;
        lockouthits(S) = AEdata(2).event(j);
        AEdata(2).event(j) = [];
    end
    break
end
end
end

% % Plotter
% for en = 1:length(lockouthits)
%     subplot(2,1,1)
%     plot(lockouthits(en).wavetime,lockouthits(en).wave)
%     title(sprintf('Bin hit %d',en))
%     % ylim([-0.05 0.05])
%     xlim([0 2e-4])
%     subplot(2,1,2)
%     plot(lockouthits(en).powerscale,lockouthits(en).power)
%     xlim([0 1.5e6])
%     pause(0.2)
% end

%% DETERMINE DEPTH OF EACH HIT
% Determines the LVDT depth at which each hit occurred. For the burst
test,
% this could be done for the pressure.

for sn = 1:2
    J = 1;
    for en = 1:length(AEdata(sn).event)
        for j = J:length(Idata.time)
            if Idata.time(j) >= AEdata(sn).event(en).at
                AEdata(sn).event(en).depth = Idata.LVDT(j);
                J = j;
                break
            end
        if j == length(Idata.time)
            AEdata(sn).event(en).depth = Idata.LVDT(end);
        end
    end
end
end

%% DATA EXTRACTION
% Groups individual hits parameters into vectors for ease of analysis.

```

```

AEdata = DataExtraction(AEdata,Idata,testparam);

%% CSS
% Creates a cumulative signal strength vector by summing individual
% strengths up to each time stamp.

for sn = 1:2
    for i = 1:length(AEdata(sn).Ihits)
        if AEdata(sn).Ihits(i) >= 1
            AEdata(sn).CSS(i) =
sum(AEdata(sn).all.CSS(1:AEdata(sn).Ihits(i)));
        else
            AEdata(sn).CSS(i) = 0;
        end
    end
end

%% IA
% Determines the historic index and the severity for a set number of
% evenly spaced points.

for sn = 1:2
    numpoints = length(AEdata(sn).event);
    AEdata(sn).HI = 0; AEdata(sn).SR = 0;
    [AEdata(sn).HI,AEdata(sn).SR] =
IntensityAnalysis(AEdata(sn).all.CSS,numpoints);
end

%% CREATE BINS
% This code could seperate out a specific group of hits by specifying
% one
% or more criteria and grouping the hits that fit those criteria.

clear bin;
for sn = 1:length(AEdata)

    % Large amplitude, Large duration
    criterial1 = AEdata(sn).norm.decamp >= 0.8;
    criteria2 = AEdata(sn).norm.durations >= 0.8;
    criteria = criterial1.*criteria2;
    bin(1).AEdata(sn) = CreateBin(AEdata(sn),criteria,Idata,testparam);

    % Low Frequency Peak
    if sn == 1
        criteria = AEdata(sn).all.peakfreqs <= 1e5;
    else

```

```

        criteria = AEdata(sn).all.peakfreqs <= 4e5;
    end
    bin(2).AEdata(sn) = CreateBin(AEdata(sn), criteria, Idata, testparam);

    % Large rise times, short durations
    criterial1 = AEdata(sn).norm.risetimes >= 0.2;
    criteria2 = AEdata(sn).norm.durations <= 0.9;
    criteria = criterial1.*criteria2;
    bin(3).AEdata(sn) = CreateBin(AEdata(sn), criteria, Idata, testparam);

    % Large Amps
    criteria = AEdata(sn).all.decamps >= 90;
    bin(4).AEdata(sn) = CreateBin(AEdata(sn), criteria, Idata, testparam);

    %
    %     % Large amp, early in test
    %     criterial1 = AEdata(sn).all.decamps >= 68;
    %     criteria2 = AEdata(sn).all.depths <= 4;
    %     criteria = criterial1.*criteria2;
    %     bin(8).AEdata(sn) =
    CreateBin(AEdata(sn), criteria, Idata, testparam);
end

%% K-MEANS
% Performs a k-means clustering algorithm for a specific k and X.

clear means;

% Sensor 1
sn = 1; k = 3; clear X;
X(:,1) = AEdata(sn).norm.decamps;
X(:,2) = AEdata(sn).norm.durations;
X(:,3) = AEdata(sn).norm.risetimes;
[means(sn).clustering, means(sn).C] = kmeans(X, k);

% Sensor 2
sn = 2 ; k = 3; clear X;
X(:,1) = AEdata(sn).norm.decamps;
X(:,2) = AEdata(sn).norm.durations;
X(:,3) = AEdata(sn).norm.risetimes;
[means(sn).clustering, means(sn).C] = kmeans(X, k);

for sn = 1:2
    % means(sn).cluster = 0;
    for i = 1:k
        S=0;
        for en = 1:length(means(sn).clustering)
            if means(sn).clustering(en) == i
                S=S+1;
                cluster(i).event(S) = AEdata(sn).event(en);
            end
        end
    end
end

```

```

end
means(sn).cluster = cluster;
clear cluster;
means(sn).cluster =
DataExtraction(means(sn).cluster,Idata,testparam);
end

% Get clusters to match
% means(2).cluster(4) = means(2).cluster(3);
% means(2).cluster(3) = means(2).cluster(2);
% means(2).cluster(2) = means(2).cluster(4);
% means(2).cluster(4) = [];

%% PLOTS

% Hits vs LVDT
figure(2)
plot(Idata.LVDT,AEdata(1).Ihits,'b',Idata.LVDT,AEdata(2).Ihits,'r');
xlabel('LVDT Depth (in)')
ylabel('Cumulative Hits')
legend('Sensor 1','Sensor 2','location','NorthWest')

% Instrumentation
% Idata.load = -(Idata.load-Idata.load(1));
[ax,h1,h2] = plotyy(Idata.LVDT,Idata.load,[Idata.LVDT' Idata.LVDT'
Idata.LVDT' Idata.LVDT'],[Idata.SGL1' Idata.SGL2' (Idata.SGH1./3)'
Idata.SGH2']);
set(ax(1),'ylim',[-5 50])
set(ax(2),'ylim',[-100 2000])
set(ax(1),'ytick',[0 25 50])
set(ax(2),'ytick',[0 1000 2000])
ylabel(ax(1),'Load(k)')
ylabel(ax(2),'Strain (microstrain)')
xlim(ax(1),[0 max(Idata.LVDT)])
xlim(ax(2),[0 max(Idata.LVDT)])
xlabel('LVDT Depth (in)')
legend([h1;h2],'Load','SGL1','SGL2','SGH1','SGH2','location','northwest')

% Event Depths and Amplitudes
figure('units','normalized','position',[.1 .1 .5 .3])
for sn = 1:2
    subplot(1,2,sn)
    plot(AEdata(sn).all.depths,AEdata(sn).all.decamps,'o');
    xlabel('Dent Depth (in)')
    ylabel('Amplitude (dB)')
    title(sprintf('Sensor %d',sn))
    xlim([0 max(Idata.LVDT)])

```

```

        ylim([40 100])
end

% CSS
plot(Idata.LVDT,AEdata(1).CSS,'b',Idata.LVDT,AEdata(2).CSS,'r');
xlabel('LVDT Depth (in)')
ylabel('Cumulative Signal Strength (relative)')
legend('Sensor 1','Sensor 2','location','NorthWest')

% Intensity Analysis Seperate
for sn = 1:2
    hitsx = 1:length(AEdata(sn).event);
    subplot(1,2,sn)
    [ax,h1,h2] = plotyy(hitsx,AEdata(sn).HI,hitsx,AEdata(sn).SR);
    set(h1,'color',[0 0 1])
    set(h2,'color',[0 1 0])
    ylim(ax(1),[0.25 1.75])
    ylim(ax(2),[0 26])
    set(ax(1),'ytick',[0.25 1 1.75])
    set(ax(2),'ytick',[0 13 26])
    xlim(ax(1),[0 hitsx(end)])
    xlim(ax(2),[0 hitsx(end)])
    title(sprintf('Sensor %d',sn))
    xlabel('Cumulative Hits')
    if sn == 1
        ylabel(ax(1),'Historic Index')
    else
        ylabel(ax(2),'Severity')
    end
end
end
figure('units','normalized','position',[.1 .1 .5 .3])

% Intensity Analysis together
plot(AEdata(1).HI,AEdata(1).SR,'b',AEdata(2).HI,AEdata(2).SR,'r')
xlabel('Historic Index')
ylabel('Severity')
legend('Sensor 1','Sensor 2','location','northwest')

% Intensity Analysis Dots
plot(AEdata(1).HI,AEdata(1).SR,'bo',AEdata(2).HI,AEdata(2).SR,'ro')
xlabel('Historic Index')
ylabel('Severity')
legend('Sensor 1','Sensor 2','location','northwest')

% Amplitudes vs Durations
figure('units','normalized','position',[.1 .1 .5 .3])
for sn = 1:2
    subplot(1,2,sn)
    plot(AEdata(sn).norm.deamps,AEdata(sn).norm.durations,'o');
    xlabel('Normalized Amplitude')
    ylabel('Normalized Duration')
    title(sprintf('Sensor %d',sn))
end

```



```

        %xlim([0 max(Idata.LVDT)])
        % ylim([0 2e-4])
end

% Amplitudes vs Rise Times
figure('units','normalized','position',[.1 .1 .5 .3])
for sn = 1:2
    subplot(1,2,sn)
    plot(AEdata(sn).norm.deamps,AEdata(sn).norm.risetimes,'o');
    xlabel('Normalized Amplitude')
    ylabel('Normalized Rise Time')
    title(sprintf('Sensor %d',sn))
    %xlim([0 max(Idata.LVDT)])
    %ylim([0 2e-4])
end

% Amplitudes vs Frequency Peaks
figure('units','normalized','position',[.1 .1 .5 .3])
for sn = 1:2
    subplot(1,2,sn)
    plot(AEdata(sn).norm.deamps,AEdata(sn).all.peakfreqs,'o');
    xlabel('Normalized Amplitude')
    ylabel('Peak Frequency (Hz)')
    title(sprintf('Sensor %d',sn))
    %xlim([0 max(Idata.LVDT)])
    ylim([0 1e6])
end

% Amplitudes vs Frequency Centroids
figure('units','normalized','position',[.1 .1 .5 .3])
for sn = 1:2
    subplot(1,2,sn)
    plot(AEdata(sn).norm.deamps,AEdata(sn).all.freqcents,'o');
    xlabel('Normalized Amplitude')
    ylabel('Frequency Centroid (Hz)')
    title(sprintf('Sensor %d',sn))
    %xlim([0 max(Idata.LVDT)])
    ylim([0 1e6])
end

% Durations vs Rise Times
figure('units','normalized','position',[.1 .1 .5 .3])
for sn = 1:2
    subplot(1,2,sn)
    plot(AEdata(sn).norm.durations,AEdata(sn).norm.risetimes,'o');
    xlabel('Normalized Durations')
    ylabel('Normalized Rise Times')
    title(sprintf('Sensor %d',sn))
    %xlim([0 max(Idata.LVDT)])
    % ylim([0 1e6])
end

```

```

% Event Depths vs Rise Times
figure('units','normalized','position',[.1 .1 .5 .3])
for sn = 1:2
    subplot(1,2,sn)
    plot(AEdata(sn).all.depths,AEdata(sn).norm.risetimes,'o');
    xlabel('Dent Depth (in)')
    ylabel('Normalized Rise Time')
    title(sprintf('Sensor %d',sn))
    xlim([0 max(Idata.LVDT)])
    %ylim([50 110])
end

% Event Depths vs Durations
figure('units','normalized','position',[.1 .1 .5 .3])
for sn = 1:2
    subplot(1,2,sn)
    plot(AEdata(sn).all.depths,AEdata(sn).norm.durations,'o');
    xlabel('Dent Depth (in)')
    ylabel('Normalized Durations')
    title(sprintf('Sensor %d',sn))
    xlim([0 max(Idata.LVDT)])
    %ylim([50 110])
end

% Event Depths vs Peak Frequencies
figure('units','normalized','position',[.1 .1 .5 .3])
for sn = 1:2
    subplot(1,2,sn)
    plot(AEdata(sn).all.depths,AEdata(sn).all.peakfreqs,'o');
    xlabel('Dent Depth (in)')
    ylabel('Normalized Durations')
    title(sprintf('Sensor %d',sn))
    xlim([0 max(Idata.LVDT)])
    ylim([0 1e6])
end

% Individual Hit Plot
sn = 1; en = 1805;
plot(AEdata(sn).event(en).wavetime,AEdata(sn).event(en).wave)
xlabel('Time (s)')
ylabel('Amplitude (V)')
xlim([0 2e-4])
ylim([-4 4])
% ylim([-0.1 0.1])
subplot(1,2,2)
plot(AEdata(sn).event(en).powerscale,AEdata(sn).event(en).power)
xlim([0 1e6])

% Matching Hits Plots
en1 = 800; en2 = 5350;
subplot(2,2,1)

```

```

plot(AEdata(1).event(en1).wavetime,AEdata(1).event(en1).wave)
subplot(2,2,2)
plot(AEdata(1).event(en1).powerscale,AEdata(1).event(en1).power)
xlim([0 1e6])
subplot(2,2,3)
plot(AEdata(2).event(en2).wavetime,AEdata(2).event(en2).wave)
subplot(2,2,4)
plot(AEdata(2).event(en2).powerscale,AEdata(2).event(en2).power)
xlim([0 1e6])

%% BIN PLOTS

% Scatter Plot
bn = 3; sn = 2;
plot(bin(bn).AEdata(sn).all.risetimes,bin(bn).AEdata(sn).all.durations,
'o');
xlabel('Sensor 1 Amplitudes (V)')
ylabel('Sensor 1 Rise Times (s)')

% Event Depths and Amplitudes
bn = 2;
figure('units','normalized','position',[.1 .1 .5 .3])
for sn = 1:2
    subplot(1,2,sn)

plot(bin(bn).AEdata(sn).all.depths,bin(bn).AEdata(sn).all.decamps,'o');
    xlabel('Dent Depth (in)')
    ylabel('Amplitude (dB)')
    title(sprintf('Sensor %d',sn))
    xlim([0 max(Idata.LVDT)])
    ylim([50 100])
end

% Plot all hits in a bin
bn = 4; sn = 2;
for en = 1:length(bin(bn).AEdata(sn).event)
    subplot(2,1,1)

plot(bin(bn).AEdata(sn).event(en).wavetime,bin(bn).AEdata(sn).event(en)
.wave)
    title(sprintf('Bin hit %d',en))
    % ylim([-0.05 0.05])
    subplot(2,1,2)

plot(bin(bn).AEdata(sn).event(en).powerscale,bin(bn).AEdata(sn).event(e
n).power)
    % xlim([0 1.5e6])
    pause(2)
end

% Event Amplitudes, Frequency Peaks

```

```

bn = 3;
figure('units','normalized','position',[.1 .1 .5 .3])
for sn = 1:2
    subplot(1,2,sn)

    plot(bin(bn).AEdata(sn).all.depths,bin(bn).AEdata(sn).all.peakfreqs,'o')
    ;
    xlabel('Dent Depth (in)')
    ylabel('Amplitude (dB)')
    title(sprintf('Sensor %d',sn))
    xlim([0 max(Idata.LVDT)])
    %     ylim([50 100])
end

% Plot an individual hit
bn = 4; sn = 2; en = 3;
figure('units','normalized','position',[.1 .1 .5 .3])
subplot(1,2,1)
plot(bin(bn).AEdata(sn).event(en).wavetime,bin(bn).AEdata(sn).event(en)
    .wave)
title(sprintf('Sensor %d Waveform,
Depth=%3.2fin',sn,bin(bn).AEdata(sn).event(en).depth))
xlabel('Time (s)')
ylabel('Amplitude (V)')
xlim([0 2e-4])
subplot(1,2,2)
plot(bin(bn).AEdata(sn).event(en).powerscale,bin(bn).AEdata(sn).event(en)
    .power)
title(sprintf('Sensor %d Spectrum,
Depth=%3.2fin',sn,bin(bn).AEdata(sn).event(en).depth))
xlabel('Frequency (Hz)')
ylabel('Amplitude (V)')
xlim([0 1e6])

% Compare 2 hits
bn = 4; sn = 2; en = 15;
figure('units','normalized','position',[.1 .1 .5 .6])
subplot(2,2,1)
plot(bin(bn).AEdata(sn).event(en).wavetime,bin(bn).AEdata(sn).event(en)
    .wave)
title(sprintf('Sensor %d Waveform,
Depth=%3.2fin',sn,bin(bn).AEdata(sn).event(en).depth))
% plot(AEdata(sn).event(en).wavetime,AEdata(sn).event(en).wave)
% title(sprintf('Sensor %d Waveform,
Depth=%3.2fin',sn,AEdata(sn).event(en).depth))
xlabel('Time (s)')
ylabel('Amplitude (V)')
xlim([0 2e-4])
subplot(2,2,2)
plot(bin(bn).AEdata(sn).event(en).powerscale,bin(bn).AEdata(sn).event(en)
    .power)
title(sprintf('Sensor %d Spectrum,
Depth=%3.2fin',sn,bin(bn).AEdata(sn).event(en).depth))

```

```

% plot(AEdata(sn).event(en).powerscale,AEdata(sn).event(en).power)
% title(sprintf('Sensor %d Spectrum,
Depth=%3.2fin',sn,AEdata(sn).event(en).depth))
xlabel('Frequency (Hz)')
ylabel('Amplitude (V)')
xlim([0 1e6])

bn = 2; sn = 1; en = 650;
subplot(2,2,3)
%
plot(bin(bn).AEdata(sn).event(en).wavetime,bin(bn).AEdata(sn).event(en)
.wave)
% title(sprintf('Sensor %d Waveform,
Depth=%3.2fin',sn,bin(bn).AEdata(sn).event(en).depth))
plot(AEdata(sn).event(en).wavetime,AEdata(sn).event(en).wave)
title(sprintf('Sensor %d Waveform,
Depth=%3.2fin',sn,AEdata(sn).event(en).depth))
xlabel('Time (s)')
ylabel('Amplitude (V)')
xlim([0 2e-4])
subplot(2,2,4)
%
plot(bin(bn).AEdata(sn).event(en).powerscale,bin(bn).AEdata(sn).event(e
n).power)
% title(sprintf('Sensor %d Spectrum,
Depth=%3.2fin',sn,bin(bn).AEdata(sn).event(en).depth))
plot(AEdata(sn).event(en).powerscale,AEdata(sn).event(en).power)
title(sprintf('Sensor %d Spectrum,
Depth=%3.2fin',sn,AEdata(sn).event(en).depth))
xlabel('Frequency (Hz)')
ylabel('Amplitude (V)')
xlim([0 1e6])

```

%% THRESHOLD PLOTS

```

% Plot an individual hit
bn = 4; sn = 2; en = 2;
figure('units','normalized','position',[.1 .1 .5 .3])
subplot(1,2,1)
plot(bin(bn).AEdata(sn).event(en).wavetime,bin(bn).AEdata(sn).event(en)
.wave)% , [0 2e-4],[0.033 0.033],'c',[0 2e-4],[0.01 0.01],'g',[0 2e-
4],[0.0056 0.0056],'r',[0 2e-4],[-0.033 -0.033],'c',[0 2e-4],[-0.01 -
0.01],'g',[0 2e-4],[-0.0056 -0.0056],'r')
title(sprintf('Sensor %d Waveform,
Depth=%3.2fin',sn,bin(bn).AEdata(sn).event(en).depth))
xlabel('Time (s)')
ylabel('Amplitude (V)')
xlim([0 2e-4])
ylim([-4 4])
subplot(1,2,2)
plot(bin(bn).AEdata(sn).event(en).wavetime,bin(bn).AEdata(sn).event(en)
.wave,[0 2e-4],[0.033 0.033],'c',[0 2e-4],[0.01 0.01],'g',[0 2e-

```

```

4],[0.0056 0.0056], 'r',[0 2e-4],[-0.033 -0.033], 'c',[0 2e-4],[-0.01 -
0.01], 'g',[0 2e-4],[-0.0056 -0.0056], 'r')
title(sprintf('Sensor %d Waveform,
Depth=%3.2fin',sn,bin(bn).AEdata(sn).event(en).depth))
xlabel('Time (s)')
ylabel('Amplitude (V)')
xlim([0 2e-4])
ylim([-0.05 0.05])
legend('Waveform','50db','40db','35db')

%% K-MEANS PLOTS

% Depths vs. Decamps
figure('units','normalized','position',[.1 .1 .5 .3])
for sn = 1:2
    subplot(1,2,sn)

    plot(means(sn).cluster(1).all.depths,means(sn).cluster(1).all.decamps, '
    ro',...

    means(sn).cluster(2).all.depths,means(sn).cluster(2).all.decamps, 'bo', .
    ..

    means(sn).cluster(3).all.depths,means(sn).cluster(3).all.decamps, 'go')
        xlabel('Dent Depth (in)')
        ylabel('Amplitude (dB)')
        title(sprintf('Sensor %d',sn))
        xlim([0 max(Idata.LVDT)])
        ylim([50 100])
    end

% Durations vs. Rise Times
figure('units','normalized','position',[.1 .1 .5 .3])
for sn = 1:2
    subplot(1,2,sn)

    plot(means(sn).cluster(1).all.durations,means(sn).cluster(1).all.riseti
    mes, 'ro',...

    means(sn).cluster(2).all.durations,means(sn).cluster(2).all.risetimes, '
    bo',...

    means(sn).cluster(3).all.durations,means(sn).cluster(3).all.risetimes, '
    go')
        xlabel('Durations (s)')
        ylabel('Rise Times (s)')
        title(sprintf('Sensor %d',sn))
        if sn == 1
            lim = 2.04e-4;
        else
            lim = 2e-4;
        end
end

```

```

        xlim([0 lim])
        ylim([0 lim])
end

% Decamps vs. Durations
figure('units','normalized','position',[.1 .1 .5 .3])
for sn = 1:2
    subplot(1,2,sn)

    plot(means(sn).cluster(1).all.decamps,means(sn).cluster(1).all.durations, 'ro', ...

    means(sn).cluster(2).all.decamps,means(sn).cluster(2).all.durations, 'bo', ...

    means(sn).cluster(3).all.decamps,means(sn).cluster(3).all.durations, 'go', ...
        xlabel('Amplitudes (dB)')
        ylabel('Durations (s)')
        title(sprintf('Sensor %d',sn))
        if sn == 1
            lim = 2.04e-4;
        else
            lim = 2e-4;
        end
        ylim([0 lim])
end

```

A.2 – SetTest.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% SetTest.m %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% By Jeffrey Michel
% Texas A&M University
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Sets the test parameters based on the specified test name.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
function [Idata,testparam] = SetTest(whichtest)

switch whichtest
case 'BurstTest'
    Idata = BurstTestExcelCorrector;
    testparam.whichtest = whichtest;
    testparam.testtype = 'Pipe';
    testparam.numsensors = 2;
    testparam.numsteps = 2;
    testparam.hits(1,1) = 1759;
    testparam.hits(1,2) = 3;
    testparam.hits(2,1) = 4659;
    testparam.hits(2,2) = 522;
    testparam.timecorrector(1) = 441.3;
    testparam.timecorrector(2) = 4638.2;
    testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\BurstTest1';
    testparam.stepname{1} = 'BurstTestStep2';
    testparam.stepname{2} = 'BurstTestStep3';
    testparam.threshold = 0.01;
    testparam.filter(1) = 100;
    testparam.filter(2) = 100;

case 'DentTest1'
    Idata = Pipe9ExcelCorrector;
    testparam.whichtest = whichtest;
    testparam.testtype = 'Pipe';
    testparam.numsensors = 2;
    testparam.numsteps = 2;
    testparam.hits(1,1) = 1264;
    testparam.hits(1,2) = 2653;
    testparam.hits(2,1) = 894;
    testparam.hits(2,2) = 2665;
    testparam.timecorrector(1) = 140;
    testparam.timecorrector(2) = 1.343409e+04;
    testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\DentTest1 Waveforms\F-40db Waveforms';
    testparam.stepname{1} = 'DentTestStep2F-40db';
    testparam.stepname{2} = 'DentTestStep3F-40db';
    testparam.threshold = 0.01;
    testparam.filter(1) = 100;
    testparam.filter(2) = 100;
```



```

case 'DentTest2'
    Idata = Pipe4ExcelCorrector;
    testparam.whichtest = whichtest;
    testparam.testtype = 'Pipe';
    testparam.numsensors = 2;
    testparam.numsteps = 4;
    testparam.hits(1,1) = 137;
    testparam.hits(1,2) = 13;
    testparam.hits(1,3) = 156;
    testparam.hits(1,4) = 27;
    testparam.hits(2,1) = 336;
    testparam.hits(2,2) = 36;
    testparam.hits(2,3) = 588;
    testparam.hits(2,4) = 57;
    testparam.timecorrector(1) = 175;
    testparam.timecorrector(2) = 4138+175;
    testparam.timecorrector(3) = 11281+175;
    testparam.timecorrector(4) = 76623+175;
    testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\DentTest2 Waveforms';
    testparam.stepname{1} = 'DentTest2Step1';
    testparam.stepname{2} = 'DentTest2Step2';
    testparam.stepname{3} = 'DentTest2Step3';
    testparam.stepname{4} = 'DentTest2Step4';
    testparam.threshold = 0.01;
    testparam.filter(1) = 100;
    testparam.filter(2) = 100;

case 'DentTest3'
    Idata = Pipe2ExcelCorrector;
    testparam.whichtest = whichtest;
    testparam.testtype = 'Pipe';
    testparam.numsensors = 2;
    testparam.numsteps = 2;
    testparam.hits(1,1) = 742;
    testparam.hits(1,2) = 196;
    testparam.hits(2,1) = 8341;
    testparam.hits(2,2) = 2160;
    testparam.timecorrector(1) = 0;
    testparam.timecorrector(2) = 23383;
    testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\DentTest3 Waveforms';
    testparam.stepname{1} = 'DentTest3Step1F-40db';
    testparam.stepname{2} = 'DentTest3Step2F-40db';
    testparam.threshold = 0.01;
    testparam.filter(1) = 100;
    testparam.filter(2) = 400;

case 'DentTest3-50db'
    Idata = Pipe2ExcelCorrector;
    testparam.whichtest = whichtest;
    testparam.testtype = 'Pipe';

```

```

testparam.numsensors = 2;
testparam.numsteps = 2;
testparam.hits(1,1) = 668;
testparam.hits(1,2) = 181;
testparam.hits(2,1) = 1412;
testparam.hits(2,2) = 265;
testparam.timecorrector(1) = 0;
testparam.timecorrector(2) = 23383;
testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\DentTest3 50db Waveforms';
testparam.stepname{1} = 'DentTest3Step1F-50db';
testparam.stepname{2} = 'DentTest3Step2F-50db';
testparam.threshold = 0.01;
testparam.filter(1) = 100;
testparam.filter(2) = 400;

case 'DummyTensileCheck1'
    Idata = 0;
    testparam.whichtest = whichtest;
    testparam.testtype = 'Tensile';
    testparam.numsensors = 1;
    testparam.numsteps = 1;
    testparam.hits(1,1) = 284;
    testparam.timecorrector(1) = 0;
    testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\Dummy Tensile Waveforms 042215\Dummy Test 3';
    testparam.stepname{1} = 'DummyTensileTest3';
    testparam.threshold = 0.01;
    testparam.filter(1) = 100;
    testparam.filter(2) = 100;

case 'DummyTensileCheck2'
    Idata = 0;
    testparam.whichtest = whichtest;
    testparam.testtype = 'Tensile';
    testparam.numsensors = 1;
    testparam.numsteps = 1;
    testparam.hits(1,1) = 3675;
    testparam.timecorrector(1) = 0;
    testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\Dummy Tensile Waveforms 042215\Dummy Test 4';
    testparam.stepname{1} = 'DummyTensileTest4';
    testparam.threshold = 0.01;
    testparam.filter(1) = 100;
    testparam.filter(2) = 100;

```

```

%%%%%%%%%%%%%% CALIBRATION TESTS %%%%%%%%%%%%%%%

```

```

case 'SensorSimilarity'

```

```

Idata = 0;
testparam.whichtest = whichtest;
testparam.testtype = 'Calibration';
testparam.numsensors = 2;
testparam.numsteps = 1;
testparam.hits(1,1) = 10;
testparam.hits(2,1) = 10;
testparam.timecorrector(1) = 0;
testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\Calibration Waveforms';
testparam.stepname{1} = 'SensorSimilarity6';
testparam.threshold = 0.01;
testparam.filter(1) = 100;
testparam.filter(2) = 100;

case 'SensorSimilarity2'
Idata = 0;
testparam.whichtest = whichtest;
testparam.testtype = 'Calibration';
testparam.numsensors = 2;
testparam.numsteps = 1;
testparam.hits(1,1) = 10;
testparam.hits(2,1) = 10;
testparam.timecorrector(1) = 0;
testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\Calibration Waveforms';
testparam.stepname{1} = 'SensorSimilarity9';
testparam.threshold = 0.01;
testparam.filter(1) = 100;
testparam.filter(2) = 100;

case 'SensorSimilarityFlat'
Idata = 0;
testparam.whichtest = whichtest;
testparam.testtype = 'Calibration';
testparam.numsensors = 2;
testparam.numsteps = 1;
testparam.hits(1,1) = 10;
testparam.hits(2,1) = 10;
testparam.timecorrector(1) = 0;
testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\Calibration Waveforms';
testparam.stepname{1} = 'SensorSimilarityFlat5';
testparam.threshold = 0.01;
testparam.filter(1) = 100;
testparam.filter(2) = 100;

case 'SetupChange1'
Idata = 0;
testparam.whichtest = whichtest;
testparam.testtype = 'Calibration';
testparam.numsensors = 1;
testparam.numsteps = 1;

```

```

testparam.hits(1,1) = 6;
testparam.timecorrector(1) = 0;
testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\Calibration Waveforms';
testparam.stepname{1} = 'SetupChange1';
testparam.threshold = 0.01;
testparam.filter(1) = 100;
testparam.filter(2) = 100;

case 'SetupChange2'
    Idata = 0;
    testparam.whichtest = whichtest;
    testparam.testtype = 'Calibration';
    testparam.numsensors = 1;
    testparam.numsteps = 1;
    testparam.hits(1,1) = 5;
    testparam.timecorrector(1) = 0;
    testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\Calibration Waveforms';
    testparam.stepname{1} = 'SetupChange2';
    testparam.threshold = 0.1;
    testparam.filter(1) = 100;
    testparam.filter(2) = 100;

case 'SetupChange3'
    Idata = 0;
    testparam.whichtest = whichtest;
    testparam.testtype = 'Calibration';
    testparam.numsensors = 1;
    testparam.numsteps = 1;
    testparam.hits(1,1) = 5;
    testparam.timecorrector(1) = 0;
    testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\Calibration Waveforms';
    testparam.stepname{1} = 'SetupChange3';
    testparam.threshold = 0.01;
    testparam.filter(1) = 100;
    testparam.filter(2) = 100;

case 'SetupChange4'
    Idata = 0;
    testparam.whichtest = whichtest;
    testparam.testtype = 'Calibration';
    testparam.numsensors = 1;
    testparam.numsteps = 1;
    testparam.hits(1,1) = 6;
    testparam.timecorrector(1) = 0;
    testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\Calibration Waveforms';
    testparam.stepname{1} = 'SetupChange4';
    testparam.threshold = 0.01;
    testparam.filter(1) = 100;
    testparam.filter(2) = 100;

```

```

case 'SetupChange5'
    Idata = 0;
    testparam.whichtest = whichtest;
    testparam.testtype = 'Calibration';
    testparam.numsensors = 1;
    testparam.numsteps = 1;
    testparam.hits(1,1) = 7;
    testparam.timecorrector(1) = 0;
    testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\Calibration Waveforms';
    testparam.stepname{1} = 'SetupChange5';
    testparam.threshold = 0.01;
    testparam.filter(1) = 100;
    testparam.filter(2) = 100;

case 'WaveSpeed'
    Idata = 0;
    testparam.whichtest = whichtest;
    testparam.testtype = 'Calibration';
    testparam.numsensors = 2;
    testparam.numsteps = 1;
    testparam.hits(1,1) = 10;
    testparam.hits(2,1) = 10;
    testparam.timecorrector(1) = 0;
    testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\Calibration Waveforms';
    testparam.stepname{1} = 'WaveSpeed1';
    testparam.threshold = 0.01;
    testparam.filter(1) = 100;
    testparam.filter(2) = 100;

case 'LongiShort'
    Idata = 0;
    testparam.whichtest = whichtest;
    testparam.testtype = 'Calibration';
    testparam.numsensors = 1;
    testparam.numsteps = 1;
    testparam.hits(1,1) = 44;
    testparam.timecorrector(1) = 0;
    testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\Calibration Waveforms';
    testparam.stepname{1} = 'LongiShort2';
    testparam.threshold = 0.01;
    testparam.filter(1) = 100;
    testparam.filter(2) = 100;

case 'LongiLong'
    Idata = 0;
    testparam.whichtest = whichtest;
    testparam.testtype = 'Calibration';
    testparam.numsensors = 1;
    testparam.numsteps = 1;

```

```

testparam.hits(1,1) = 41;
testparam.timecorrector(1) = 0;
testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\Calibration Waveforms';
testparam.stepname{1} = 'LongiLong1';
testparam.threshold = 0.01;
testparam.filter(1) = 100;
testparam.filter(2) = 100;

case 'HoopLong'
    Idata = 0;
    testparam.whichtest = whichtest;
    testparam.testtype = 'Calibration';
    testparam.numsensors = 1;
    testparam.numsteps = 1;
    testparam.hits(1,1) = 37;
    testparam.timecorrector(1) = 0;
    testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\Calibration Waveforms';
    testparam.stepname{1} = 'HoopLong1';
    testparam.threshold = 0.01;
    testparam.filter(1) = 100;
    testparam.filter(2) = 100;

case 'WeldTest'
    Idata = 0;
    testparam.whichtest = whichtest;
    testparam.testtype = 'Calibration';
    testparam.numsensors = 1;
    testparam.numsteps = 1;
    testparam.hits(1,1) = 16;
    testparam.timecorrector(1) = 0;
    testparam.folder = 'C:\Users\mich2905\Desktop\Acoustic
Data\Calibration Waveforms';
    testparam.stepname{1} = 'HoopLong1';
    testparam.threshold = 0.01;
    testparam.filter(1) = 100;
    testparam.filter(2) = 100;
end

```

A.3 – BurstTestExcelCorrector.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% BurstTestExcelCorrector.m %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% By Jeffrey Michel
% Texas A&M University
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Sorts and adjusts the instrumentation data for the burst test.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function Idata = BurstTestExcelCorrector

%% Strain, load, displacement data
[~,~,spreadsheet] = xlsread('2015-03-25_Blade_Colonial_HBL50_12inch_burst.xlsx');
for i = 1:length(spreadsheet(:,1))-2
    raw.time(i) = spreadsheet{i+2,3};
    raw.pressure(i) = spreadsheet{i+2,5};
    raw.SG1(i) = spreadsheet{i+2,6};
    raw.SG2(i) = spreadsheet{i+2,7};
    raw.SG3(i) = spreadsheet{i+2,8};
    raw.SG4(i) = spreadsheet{i+2,9};
    raw.SG5(i) = spreadsheet{i+2,10};
    raw.CLP1(i) = spreadsheet{i+2,11};
    raw.CLP2(i) = spreadsheet{i+2,12};
end

Idata.time = raw.time;
Idata.pressure = raw.pressure;
Idata.SG1 = raw.SG1;
Idata.SG2 = raw.SG2;
Idata.SG3 = raw.SG3;
Idata.SG4 = raw.SG4;
Idata.SG5 = raw.SG5;
Idata.CLP1 = raw.CLP1;
Idata.CLP2 = raw.CLP2;
```

A.4 – Pipe9ExcelCorrector.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Pipe9ExcelCorrector.m %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% By Jeffrey Michel
% Texas A&M University
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Sorts and adjusts the instrumentation data for dent test 1.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function Idata = Pipe9ExcelCorrector

%% Strain, load, displacement data
[~,~,spreadsheet] = xlsread('2015-4-1 Blade Pipe 9 Denting.xlsx');
for i = 1:length(spreadsheet(:,1))-2
    raw.time(i) = spreadsheet{i+2,3};
    raw.load(i) = spreadsheet{i+2,4};
    raw.displ(i) = abs(spreadsheet{i+2,5}-3); % Adjust & flip
displacement
    raw.SGL1(i) = spreadsheet{i+2,6};
    raw.SGL2(i) = spreadsheet{i+2,7};
    raw.SGH1(i) = spreadsheet{i+2,8};
    raw.SGH2(i) = spreadsheet{i+2,9};
    raw.LVDT(i) = spreadsheet{i+2,10};
end

%%%% STOP&START INFORMATION %%%%

% Began test at 90
true.LVDT(1:90) = 0;

%%% STOP 1
% Stopped to reset LVDT, took place at 3630
% Resumed at 4093
true.LVDT(91:3630) = raw.LVDT(91:3630) - raw.LVDT(90);
true.LVDT(3631:4093) = true.LVDT(3630);

%%% STOP 2
% Stopped to scan, took place at 4784
% Resumed at 7764
true.LVDT(4094:4784) = raw.LVDT(4094:4784) - raw.LVDT(4093) +
true.LVDT(3630);
true.LVDT(4785:7764) = true.LVDT(4784);

%%% STOP 3
% Stopped at the end of day 1 at 9497
% Started the beginning of day 2 at 10600
true.LVDT(7765:9497) = raw.LVDT(7765:9497) - raw.LVDT(7764) +
true.LVDT(4784);
true.LVDT(9498:10600) = true.LVDT(9497);
```



```

%%% STOP 4
% Stopped to reset the LVDT at at 13785
% Resumed at 14628
true.LVDT(10601:13785) = raw.LVDT(10601:13785) - raw.LVDT(10600) +
true.LVDT(9497);
true.LVDT(13786:14628) = true.LVDT(13785);

%%% STOP 5
% Stopped briefly to adjust something, at 16381
% Resumed at 16747
true.LVDT(14629:16381) = raw.LVDT(14629:16381) - raw.LVDT(14628) +
true.LVDT(13785);
true.LVDT(16382:16747) = true.LVDT(16381);

%%% STOP 6
% Stopped to reset the LVDT at 18105
% Resumed at 19085
true.LVDT(16748:18105) = raw.LVDT(16748:18105) - raw.LVDT(16747) +
true.LVDT(16381);
true.LVDT(18106:19085) = true.LVDT(18105);

%%% STOP 7
% Stopped briefly to check something at 19485
% Resumed at 19950
true.LVDT(19086:19485) = raw.LVDT(19086:19485) - raw.LVDT(19085) +
true.LVDT(18105);
true.LVDT(19486:19950) = true.LVDT(19485);

%%% STOP 8
% Stopped to reset LVDT at 22488
% Resumed at 22836
true.LVDT(19951:22488) = raw.LVDT(19951:22488) - raw.LVDT(19950) +
true.LVDT(19485);
true.LVDT(22489:22836) = true.LVDT(22488);

%%% STOP 9
% Stopped briefly at 23718
% Resumed at 23925
true.LVDT(22837:23718) = raw.LVDT(22837:23718) - raw.LVDT(22836) +
true.LVDT(22488);
true.LVDT(23719:23925) = true.LVDT(23718);

%%% STOP 10
% Stopped to reset LVDT at 26205
% Resumed at 27110
true.LVDT(23926:26205) = raw.LVDT(23926:26205) - raw.LVDT(23925) +
true.LVDT(23718);
true.LVDT(26206:27110) = true.LVDT(26205);

%%% STOP 11
% Stopped for good at 28704

```

```

true.LVDT(27111:28704) = raw.LVDT(27111:28704) - raw.LVDT(27110) +
true.LVDT(26205);
true.LVDT(28705:29887) = true.LVDT(28704);

```

```

Idata.time = [raw.time(90:9497),raw.time(10600:28704)];
Idata.time(9409:end) = Idata.time(9409:end) - Idata.time(9409) +
Idata.time(9408) + 1;
Idata.load = [raw.load(90:9497),raw.load(10600:28704)];
Idata.displ = [raw.displ(90:9497),raw.displ(10600:28704)];
Idata.SGL1 = [raw.SGL1(90:9497),raw.SGL1(10600:28704)];
Idata.SGL2 = [raw.SGL2(90:9497),raw.SGL2(10600:28704)];
Idata.SGH1 = [raw.SGH1(90:9497),raw.SGH1(10600:28704)];
Idata.SGH2 = [raw.SGH2(90:9497),raw.SGH2(10600:28704)];
Idata.LVDT = [true.LVDT(90:9497),true.LVDT(10600:28704)];

```

A.5 – Pipe4ExcelCorrector.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Pipe4ExcelCorrector.m %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% By Jeffrey Michel
% Texas A&M University
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Sorts and adjusts the instrumentation data for dent test 2.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function Idata = Pipe4ExcelCorrector

%% Strain, load, displacement data
[~,~,spreadsheet] = xlsread('2015-4-30 Blade Pipe 4 Denting.xlsx');
for i = 1:length(spreadsheet(:,1))-2
    raw.time(i) = spreadsheet{i+2,3};
    raw.load(i) = spreadsheet{i+2,4};
    raw.displ(i) = abs(spreadsheet{i+2,5}-3); % Adjust & flip
displacement
    raw.SGL1(i) = spreadsheet{i+2,6};
    raw.SGL2(i) = spreadsheet{i+2,7};
    raw.SGH1(i) = spreadsheet{i+2,8};
    raw.SGH2(i) = spreadsheet{i+2,9};
    raw.LVDT(i) = spreadsheet{i+2,10};
end

%%%% STOP&START INFORMATION %%%%

% Began test at 176
true.LVDT(1:176) = 0;

%%% STOP 1
% Stopped to reset LVDT, took place at 3342
% Resumed at 3675
true.LVDT(176:3342) = raw.LVDT(176:3342) - raw.LVDT(176);
true.LVDT(3342:3675) = true.LVDT(3342);

%%% STOP 2
% Stopped to scan, took place at 4027
% Resumed at 4317
true.LVDT(3675:4027) = raw.LVDT(3675:4027) - raw.LVDT(3675) +
true.LVDT(3342);
true.LVDT(4027:4317) = true.LVDT(4027);

%%% STOP 3
% From 4792 to 4928
true.LVDT(4317:4792) = raw.LVDT(4317:4792) - raw.LVDT(4317) +
true.LVDT(4027);
true.LVDT(4792:4928) = true.LVDT(4792);
```

```

##### STOP 4
% From 5378 to 5491
true.LVDT(4928:5378) = raw.LVDT(4928:5378) - raw.LVDT(4928) +
true.LVDT(4792);
true.LVDT(5378:5491) = true.LVDT(5378);

##### STOP 5
% From 6269 to 6486
true.LVDT(5491:6269) = raw.LVDT(5491:6269) - raw.LVDT(5491) +
true.LVDT(5378);
true.LVDT(6269:6486) = true.LVDT(6269);

##### STOP 6
% From 8511 to 9807
true.LVDT(6486:8511) = raw.LVDT(6486:8511) - raw.LVDT(6486) +
true.LVDT(6269);
true.LVDT(8511:9807) = true.LVDT(8511);

##### STOP 7
% For good at 10530
true.LVDT(9807:10530) = raw.LVDT(9807:10530) - raw.LVDT(9807) +
true.LVDT(8511);
true.LVDT(10530:10835) = true.LVDT(10530);

Idata.time = [raw.time(176:8511),raw.time(9807:10530)];
Idata.time(8511:end) = Idata.time(8511:end) - Idata.time(8511) +
Idata.time(8511) + 1;
Idata.load = [raw.load(176:8511),raw.load(9807:10530)];
Idata.displ = [raw.displ(176:8511),raw.displ(9807:10530)];
Idata.SGL1 = [raw.SGL1(176:8511),raw.SGL1(9807:10530)];
Idata.SGL2 = [raw.SGL2(176:8511),raw.SGL2(9807:10530)];
Idata.SGH1 = [raw.SGH1(176:8511),raw.SGH1(9807:10530)];
Idata.SGH2 = [raw.SGH2(176:8511),raw.SGH2(9807:10530)];
Idata.LVDT = [true.LVDT(176:8511),true.LVDT(9807:10530)];

```

A.6 – Pipe2ExcelCorrector.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Pipe2ExcelCorrector.m %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% By Jeffrey Michel
% Texas A&M University
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Sorts and adjusts the instrumentation data for dent test 3.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function Idata = Pipe2ExcelCorrector

%% Strain, load, displacement data
[~,~,spreadsheet] = xlsread('2015-5-21 Blade Denting Pipe 2.xlsx');
for i = 1:length(spreadsheet(:,1))-2
    raw.time(i) = spreadsheet{i+2,3};
    raw.load(i) = spreadsheet{i+2,4};
    raw.displ(i) = abs(spreadsheet{i+2,5}-3); % Adjust & flip
displacement
    raw.SGL1(i) = spreadsheet{i+2,6};
    raw.SGL2(i) = spreadsheet{i+2,7};
    raw.SGH1(i) = spreadsheet{i+2,8};
    raw.SGH2(i) = spreadsheet{i+2,9};
    raw.LVDT(i) = spreadsheet{i+2,10};
    raw.SGH3(i) = spreadsheet{i+2,12};
    raw.SGH4(i) = spreadsheet{i+2,13};
end

% %%%% STOP&START INFORMATION %%%%
%
% % Began test at 58
% true.LVDT(1:176) = 0;

% Began test at 183
true.LVDT(1:183) = 0;

%%% STOP 1
% Stopped to reset LVDT, took place at 2158
% Resumed at 2318
true.LVDT(183:2158) = raw.LVDT(183:2158) - raw.LVDT(183);
true.LVDT(2158:2318) = true.LVDT(2158);

%%% STOP 2
% Stopped to scan, took place at 2970
% Resumed at 3007
true.LVDT(2319:2970) = raw.LVDT(2319:2970) - raw.LVDT(2319) +
true.LVDT(2318);
true.LVDT(2970:3007) = true.LVDT(2970);

%%% STOP 3
% Stopped from 4336 to 4423
```

```

true.LVDT(3008:4336) = raw.LVDT(3008:4336) - raw.LVDT(3008) +
true.LVDT(3007);
true.LVDT(4336:4423) = true.LVDT(4336);

%%% STOP 4
% Stopped from 6503 to 9675
true.LVDT(4424:6503) = raw.LVDT(4424:6503) - raw.LVDT(4424) +
true.LVDT(4423);
% Need to correct last 200 points
true.LVDT(6303:6503) = true.LVDT(6103:6303) - true.LVDT(6103) +
true.LVDT(6303);
true.LVDT(6503:9675) = true.LVDT(6503);

%%% STOPPED FOR GOOD
% At 11471
true.LVDT(9676:11471) = raw.LVDT(9676:11471) - raw.LVDT(9676) +
true.LVDT(9675);
true.LVDT(11471:11579) = true.LVDT(11471);

Idata.time = [raw.time(183:6503),raw.time(9675:11471)];
Idata.time(6504:end) = Idata.time(6504:end) - Idata.time(6504) +
Idata.time(6503) + 1;
Idata.time = Idata.time - Idata.time(1);
Idata.load = [raw.load(183:6503),raw.load(9675:11471)];
Idata.displ = [raw.displ(183:6503),raw.displ(9675:11471)];
Idata.SGL1 = [raw.SGL1(183:6503),raw.SGL1(9675:11471)];
Idata.SGL2 = [raw.SGL2(183:6503),raw.SGL2(9675:11471)];
Idata.SGH1 = [raw.SGH1(183:6503),raw.SGH1(9675:11471)];
Idata.SGH2 = [raw.SGH2(183:6503),raw.SGH2(9675:11471)];
Idata.LVDT = [true.LVDT(183:6503),true.LVDT(9675:11471)];

```

A.7 – CreateAEdata.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CreateAEdata.m %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% By Jeffrey Michel
% Texas A&M University
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Creates the data structure of sorted AE hit parameters for a specific
% test.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function AEdata = CreateAEdata(testparam,threshtype)

% Load all of the waveforms
for sn = 1:testparam.numsensors
    for stepn = 1:testparam.numsteps
        for en = 1:testparam.hits(sn,stepn)
            disp(sprintf('Step %d, Sensor %d, hit %d',stepn,sn,en))
            nums = num2str(sn);
            numh = num2str(en);
            fname =
[testparam.folder,'\ ',testparam.stepname{stepn},'_ ',nums,'_',numh,'.txt
'];
            if stepn == 1
                oen = en;
            else
                oen = sum(testparam.hits(sn,1:stepn-1)) + en;
            end
            AEdata(sn).event(oen) =
loadwaveform(fname,stepn,testparam.threshold,threshtype,testparam.filte
r(sn));
            AEdata(sn).event(oen).at = AEdata(sn).event(oen).at +
testparam.timecorrector(stepn);
            AEdata(sn).event(oen).sensor = sn;
        end
    end
end
```

A.8 – CreateBin.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CreateBin.m %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% By Jeffrey Michel
% Texas A&M University
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Creates a bin of specific AE hits for the analysis script.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function AEbin = CreateBin(AEdata,criteria,Idata,testparam)

S = 0;
for en = 1:length(AEdata.event)
    if criteria(en) == 1
        S = S+1;
        AEbin.event(S) = AEdata.event(en);
    end
end

AEbin = DataExtraction(AEbin,Idata,testparam);
```


A.9 – LoadWaveform.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% loadwaveform.m %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% By Jeffrey Michel
% Texas A&M University
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Loads a specific waveform's text file and determines all
% characteristics from its data vector.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function event =
loadwaveform(filename,stepn,threshold,threshtype,filter)

ts = 0.0000002;
th = ts*1024;
HDT = 2.5e-5;

% Load Data
x = importdata(filename);
event.raw = x.textdata(:,1);
event.sensor = 0;
event.step = stepn;

% Time of event.
rawtime = event.raw{11};
for j = 1:length(rawtime)
    if strcmp(rawtime(j),':') == 1
        event.at = str2num(rawtime(j+2:end));
        break
    end
end
clear rawtime;

% Wave Data
for j = 1:length(event.raw)-12
    event.wave(j) = str2num(event.raw{11+j});
    event.wavetime(j) = j*ts;
end
[event.amplitude,I] = max(abs(event.wave));
event.decamp = 20*log10(event.amplitude/0.01)+40;
event.timeofmax = event.wavetime(I);

% Counts
if strcmp(threshtype,'floating') == 1;
    temp = 0.05*event.amplitude;
    if temp >= threshold
        threshold = temp;
    end
end
event.threshold = threshold;
event.counts = 0;
```

```

event.countstopeak = 0;
event.countsafterpeak = 0;
event.timeoffirstcount = 0;
event.timeoflastcount = 0;
for i = 2:length(event.wave);

    % Need to account for negative max amplitudes
    if event.wave(i) > 0
        criteria = event.wave(i) > threshold && event.wave(i-1) <=
threshold;
    elseif event.wave(i) < 0
        criteria = event.wave(i) < -threshold && event.wave(i-1) >= -
threshold;
    end

    if criteria == 1
        event.counts = event.counts + 1;
        if event.counts == 1;
            event.timeoffirstcount = event.wavetime(i);
        end
        event.timeoflastcount = event.wavetime(i);
        if event.timeoflastcount <= event.timeofmax;
            event.countstopeak = event.counts;
        end
    end

    if event.wavetime(i) > event.timeofmax && event.wavetime(i) -
event.timeoflastcount >= HDT
        break
    end
end
event.countsafterpeak = event.counts - event.countstopeak;

% Times
event.risetime = event.timeofmax - event.timeoffirstcount;
event.droptime = event.timeoflastcount - event.timeofmax;
event.duration = event.timeoflastcount - event.timeoffirstcount;
event.initiationfreq = event.countstopeak/event.risetime;
event.reverberationfreq = event.countsafterpeak/event.droptime;

% Power Data
event.power = 2*abs(fft(event.wave))/length(event.wave);
event.power = event.power(1:floor(end/2));
event.powerscale = 1:1:length(event.power);
event.powerscale = event.powerscale/th;
for i = 1:length(event.power) % Zeroing power scales outside of limits
    if filter == 100;
        if event.powerscale(i) >= 7e5
            event.power(i) = 0;
        end
    elseif filter == 400;

```

```

        if event.powerscale(i) <= 3e5 || event.powerscale(i) >= 10.9e5
            event.power(i) = 0;
        end
    end
end
[~,I] = max(event.power);
event.peakfreq = event.powerscale(I);
event.freqcent = sum(event.power.*event.powerscale)/(sum(event.power));

% Energy
event.energy = trapz(event.wave.^2);
event.CSS = trapz(abs(event.wave));

```

A.10 – DataExtraction.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DataExtraction.m %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% By Jeffrey Michel
% Texas A&M University
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Separates each individual hits parameters into vectors for ease of
% analysis.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function AEdata = DataExtraction(AEdata,Idata,testparam)

% For histograms
for sn = 1:length(AEdata)
    for en = 1:length(AEdata(sn).event)
        AEdata(sn).all.at(en) = AEdata(sn).event(en).at;
        AEdata(sn).all.amplitudes(en) = AEdata(sn).event(en).amplitude;
        AEdata(sn).all.decamps(en) = AEdata(sn).event(en).decamp;
        AEdata(sn).all.peakfreqs(en) = AEdata(sn).event(en).peakfreq;
        AEdata(sn).all.freqcents(en) = AEdata(sn).event(en).freqcent;
        AEdata(sn).all.durations(en) = AEdata(sn).event(en).duration;
        AEdata(sn).all.risetimes(en) = AEdata(sn).event(en).risetime;
        AEdata(sn).all.counts(en) = AEdata(sn).event(en).counts;
        AEdata(sn).all.countstopeaks(en) =
AEdata(sn).event(en).countstopeak;
        AEdata(sn).all.energies(en) = AEdata(sn).event(en).energy;
        AEdata(sn).all.CSS(en) = AEdata(sn).event(en).CSS;
        AEdata(sn).all.totalenergy(en) =
sum(AEdata(sn).all.energies(1:en));
        AEdata(sn).all.initiationfreqs(en) =
AEdata(sn).event(en).initiationfreq;
        AEdata(sn).all.reverberationfreqs(en) =
AEdata(sn).event(en).reverberationfreq;
        if strcmp(testparam.testtype,'Pipe') == 1;
            if strcmp(testparam.whichtest,'BurstTest') == 1;
                AEdata(sn).all.pressures(en) =
AEdata(sn).event(en).pressure;
            else
                AEdata(sn).all.depths(en) = AEdata(sn).event(en).depth;
            end
        end
    end
end

% Normalization
for sn = 1:length(AEdata)
    AEdata(sn).norm.amplitudes =
NormalizeVector(AEdata(sn).all.amplitudes);
    AEdata(sn).norm.decamps = NormalizeVector(AEdata(sn).all.decamps);
    AEdata(sn).norm.freqcents =
NormalizeVector(AEdata(sn).all.freqcents);
```

```

    AEdata(sn).norm.peakfreqs =
NormalizeVector(AEdata(sn).all.peakfreqs);
    AEdata(sn).norm.durations =
NormalizeVector(AEdata(sn).all.durations);
    AEdata(sn).norm.risetimes =
NormalizeVector(AEdata(sn).all.risetimes);
    AEdata(sn).norm.counts = NormalizeVector(AEdata(sn).all.counts);
    AEdata(sn).norm.countstopeaks =
NormalizeVector(AEdata(sn).all.countstopeaks);
    AEdata(sn).norm.energies =
NormalizeVector(AEdata(sn).all.energies);
    AEdata(sn).norm.CSS = NormalizeVector(AEdata(sn).all.CSS);
    AEdata(sn).norm.initiationfreqs =
NormalizeVector(AEdata(sn).all.initiationfreqs);
    AEdata(sn).norm.reverberationfreqs =
NormalizeVector(AEdata(sn).all.reverberationfreqs);
    if strcmp(testparam.testtype, 'Pipe') == 1;
        if strcmp(testparam.whichtest, 'BurstTest') == 1;
            AEdata(sn).norm.pressures =
NormalizeVector(AEdata(sn).all.pressures);
        else
            AEdata(sn).norm.depths =
NormalizeVector(AEdata(sn).all.depths);
        end
    end
end

% IHits Vector
for sn = 1:length(AEdata)
    if strcmp(testparam.testtype, 'Pipe') == 1
        AEdata(sn).Ihits = IhitsVector(AEdata(sn), Idata);
    end
end

```

A.11 – IHitsVector.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% IHitsVector.m %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% By Jeffrey Michel
% Texas A&M University
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Determines the cumulative number of AE hits for each instrumentation
% timestamp.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function Ihits = IhitsVector(AEdata,Idata)

J = 1; S = 0;
for i = 1:length(AEdata.all.at)
    for j = J:length(Idata.time)
        if AEdata.all.at(i) <= Idata.time(j)
            J = j-1; S = S+1;
            break
        end
        Ihits(j) = S;
    end
end
end
Q = length(Idata.time) - length(Ihits);
if Q > 0
    for i = 1:Q
        Ihits(end+1) = Ihits(end);
    end
end
end
```

A.12 – IntensityAnalysis.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% IntensityAnalysis.m %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% By Jeffrey Michel
% Texas A&M University
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Performs an intensity analysis for a CSS vector.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [HI,SR] = IntensityAnalysis(CSS,numpoints);

Nstep = floor(length(CSS)/numpoints);

for i = 1:numpoints
    N(i) = length(CSS)-(numpoints-i)*Nstep;

    % Historic Index
    SOI = CSS(1:N(i));
    if N(i) <= 15
        K(i) = 0;
    elseif N(i) <= 16 && N(i) >= 75
        K(i) = N(i)-15;
    else
        K(i) = floor(0.8*N(i));
    end
    HI(i) = (N(i)/(N(i)-K(i))) * sum(SOI(K(i)+1:N(i)))/sum(SOI);

    % Severity
    SOM = sort(SOI,'descend');
    if N(i) < 10
        SR(i) = 0;
    else
        SR(i) = 0.1*sum(SOM(1:10));
    end
end
```

A.13 – NormalizeVector.m

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% NormalizeVector.m %%%%%%%%%%
% By Jeffrey Michel
% Texas A&M University
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Normalizes a characteristics vector for K-means algorithm.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function normal = NormalizeVector(data)

small = min(data);
large = max(data);
span = large-small;
normal = (data-small)./span;
```